

Attorney Docket No. 211748-00017

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

Applicant(s): Spurgat et al )

Serial No.: 09/858,415 )

Filed: May 16, 2001 )

Title: Proximity Synchronization of Audio  
Content Among Multiple Playback and Storage  
Devices )

Group Art )

Unit: 2682 )

Examiner: Tran, Tuan )

Confirmation No. 2468 )

**Declaration of John S. Paniaguas**  
**under 37 CFR 1.131**Mail Stop Amendment  
Commissioner for Patents  
P.O. Box 1450  
Washington, D.C. 20231

Dear Sir:

I, John S. Paniaguas, pursuant to 37 C.F.R. §1.131, declare as follows:

1. I am the attorney of record in the above-identified patent application and was an attorney of record for this case at all times relevant.
2. None of the inventors are currently with the company.

**BEST AVAILABLE COPY**

Attorney Docket No.: 211748-00017

3. A copy of an invention disclosure<sup>1</sup> for the above identified case dated February 22, 2001 is enclosed as Exhibit 1, illustrating a conception of the invention at least as early as February 22, 2001
4. Information in my prosecution file for the above identified case, kept in the normal course of business, indicates that the effective date of the above-identified patent is at least as early as March 16, 2001. In particular, I worked with the inventors to prepare a draft patent application for the above identified case. A draft of the patent application was sent to Mr. Jeff Spurgat, one of the inventors, for review by way of a letter dated March 16, 2001, attached as Exhibit 2. A copy of that application<sup>2</sup> is attached as Exhibit 3. The Examiner will kindly note the footer on each page of the draft application which illustrates a date of March 16, 2001. It is hornbook law that a patent application is constructive reduction to practice of an invention.
5. The relatively short time period from the date of the invention disclosure (i.e. February 22, 2001) to the draft application on March 16, 2001 which is only 3 days prior to the effective date of the Ross et al reference (i.e. May 19, 2001), demonstrates the Applicant's due diligence from the Applicant's conception date to the earliest effective date of the reference.
6. The undersigned, being warned that willful false statements and the like are punishable by fine or imprisonment, or both (18 U.S.C. §1001) and may jeopardize the validity of the application or any patent issuing thereon, hereby declares that the above statements made of my own knowledge are true and that all statements made on information and belief are believed to be true.

---


<sup>1</sup> The drawings have been omitted in order to reduce the bulk of this submittal. However, the Applicant is more than happy to submit the drawings if the Examiner so desires.

<sup>2</sup> See note 1

Attorney Docket No.: 211748-00017

FURTHER DECLARANT SAYETH NOT.

Date: May 15, 2006.

  
John S. Paniaguas  
Reg. No. 31,051



Chicago  
Los Angeles  
New York  
Washington, DC

525 West Monroe Street  
Suite 1600  
Chicago, IL 60661-3693  
Tel 312 902 5200  
Fax 312 902 1061  
www.kmz.com

Writer's Direct Numbers  
312 902 5312  
312 577 4532 (FAX)  
johnp@KMZ.com

March 16, 2001

**Via Hand Delivery**

Mr. Jeff Spurgat  
FULL AUDIO CORPORATION  
247 South State Street  
Suite 1220  
Chicago, IL 60604

Re: Proximity Synchronization of Audio Content Among  
Multiple Playback and Storage Devices  
Inventor(s): Spurgat, et al.  
Our File No. 11748/17

Dear Jeff:

Enclosed please find a draft patent application, informal drawings, an Declaration and Assignment prepared for the above-identified case. Please note, that I have requested some additional information throughout the application and will need that information before the application is filed in the United States Patent and Trademark Office.

Please review the application to make sure that it adequately and satisfactorily sets forth the features of the invention. Please have the co-inventor, Mr. Hoyet H. Andrews III, also review the application. Please send all your comments to me so that we can finalize the application.

Should you have any questions, please do not hesitate to contact me.

Very truly yours,

A handwritten signature in black ink, appearing to read "John".

John S. Paniaguas

JSP:mm  
Enclosures



"Jeff Spurgat" <jspurgat@fullaudio.com> on 02/22/2001 04:22:45 PM

To: "John Paniaguas" <johnp@kmz.com>

cc:

Subject: patent application #1004

John,

Attached is the patent application for "Proximity Synchronization of Audio Content Among Multiple Playback and Storage Devices". If you have any questions let me know.

Thanks,  
Jeff



- Diagrams\_1004\_v101.zip



- FAPatentApp\_1004\_v101.doc



## **FULLAUDIO PATENT APPLICATION**

### **PROXIMITY SYNCHRONIZATION OF AUDIO CONTENT AMONG MULTIPLE PLAYBACK AND STORAGE DEVICES**

*v1.0.1*

*February 22, 2001*

[illegible]

## INVENTORS

**Jeffrey Jonathan Spurgat**  
7117 New Washburn Way  
Madison, WI 53719  
County: Dane  
Citizenship: USA

**Hoyet Harrison Andrews III**  
1202 Fontaine Drive  
San Antonio, TX 78219  
County: Bexar  
Citizenship: USA



## REFERENCES

## INTRODUCTION

There are many different devices for mobile digital audio playback, such as handheld players, and in-dash automotive players. There are also fixed digital audio playback devices, such as stand-alone players and rack players that connect to a home stereo system. However, these digital audio players operate independently and without interaction with other devices. The only exception to this is typically for occasional direct-wired connection of the integrated or removable storage of the digital audio player to a computing platform for download of digital audio content. The invention described here encompasses wireless communication, interaction, and synchronization between a computing platform and various mobile or fixed digital audio players as well as among the digital audio players themselves. The computing platform may act, as part of the invention described here and through a wireless communication interface, to control the digital audio players, as a cache of digital audio data for the digital audio players, and as a gateway to the Internet for the digital audio players to access additional digital audio content and other information. The computing platform may also act, as part of the invention described here, to automatically update digital audio content on the digital audio players, to synchronize digital audio content and playlists between the digital audio players, and to automatically continue a particular playlist as the user moves from one digital audio player to another.

## DIAGRAMS

Figure 1. *Overview*

[See Fig001\_Overview.PDF]

Figure 2. *Audio Gateway*

[See Fig002\_AudioGateway.PDF]

Figure 3. *Local Wireless Network*

[See Fig003\_WirelessNetwork.PDF]

Figure 4. *Personal Computer and Digital Audio Players Configuration*

[See Fig004\_PCAndPlayers.PDF]

Figure 5. *Set-top Box and Digital Audio Players Configuration*

[See Fig005\_SetTopAndPlayers.PDF]

Figure 6. *Stand-alone Gateway and Digital Audio Players Configuration*

[See Fig006\_GatewayAndPlayers.PDF]

Figure 7. *Communication Between Digital Audio Players*

[See Fig007\_PlayerComm.PDF]

Figure 8. *Computing Platform*

[See Fig008\_ComputingPlatform.PDF]

Figure 9. *Handheld Player*

[See Fig009\_HandheldPlayer.PDF]

Figure 10. *Automotive Player*

[See Fig010\_AutomotivePlayer.PDF]

Figure 11. *Rack Player*

[See Fig011\_RackPlayer.PDF]

Figure 12. *Stand-alone Player*

[See Fig012\_StandAlonePlayer.PDF]

Figure 13. *Audio Gateway Message Handling Flow*

[See Fig013\_GatewayMessageFlow.PDF]

Figure 14. *Audio Gateway Discovery Flow*

[See Fig014\_GatewayDiscoveryFlow.PDF]

Figure 15. ***Audio Gateway Dropout Detection Flow***

[See Fig015\_GatewayDropOutFlow.PDF]

Figure 16. ***Audio Gateway Content Synchronization Flow***

[See Fig016\_GatewayContentSyncFlow.PDF]

Figure 17. ***Audio Gateway Playlist Continuation Flow***

[See Fig017\_GatewayContinuationFlow.PDF]

Figure 18. ***Player Message Handling Flow***

[See Fig018\_PlayerMessageFlow.PDF]

Figures 19-20. ***Player Discovery Flow***

[See Fig019\_PlayerDiscoveryFlowA.PDF and Fig020\_PlayerDiscoveryFlowB.PDF]

Figure 21. ***Player Dropout Detection Flow***

[See Fig021\_PlayerDropOutFlow.PDF]

Figure 22. ***Player Content Synchronization Flow***

[See Fig022\_PlayerContentSyncFlow.PDF]

Figures 23-24. ***Player Playlist Continuation Flow***

[See Fig023\_PlayerContinuationFlowA.PDF and  
Fig024\_PlayerContinuationFlowB.PDF]

## DESCRIPTION

### *Overview*

In a general configuration for the invention, illustrated in figure 1, there is a computing platform 103 connected to the Internet or other computer network 102. The computing platform 103 accesses servers 100 on the Internet or other computer network 102 that contain digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, among other things. Though the computing platform 103 can act as a digital audio player by itself, its purpose with respect to this invention is as an audio gateway for a home or business, caching digital audio content and other information 101 from servers 100 that are connected to the computing platform 103 through the Internet or other computer network 102. Using a wireless network or proprietary wireless communication 104, the computing platform 103 communicates with mobile digital audio players 115 and fixed digital audio players 116 that are within range of the wireless network or proprietary wireless communication 104. The computing platform 103 may automatically, or upon request, copy, add or remove digital audio content and other information 101, such as playlists, on mobile digital audio players 115 and fixed digital audio players 116. The digital audio content and other information 101 may be cached locally on the computing platform 103 or come from a server 100 that is accessed by the computing platform 103 over the Internet or other computer network 102. The computing platform 103 may also control mobile digital audio players 115 and fixed digital audio players 116 by changing the current playlist or the currently playing digital audio content, among other things, on the mobile digital audio players 115 or fixed digital audio players 116.

### *Audio Gateway*

A computing platform 103, as part of this invention, acts as an audio gateway for a home or business. What this means is that the computing platform 103 accesses digital audio content and other information 101 on servers 100 that are connected to the computing platform 103 through the Internet or other computer network 102. The computing platform 103 is then able to cache this digital audio content and other information 101 for distribution to mobile digital audio players 115 and fixed digital audio players 116. In addition, as part of this invention the computing platform 103, acting as an audio gateway, can synchronize content and playlists among mobile digital audio players 115 and fixed digital audio players 116. The computing platform 103 can also control mobile digital audio players 115 and fixed digital audio players 116, such as changing the current playlist or the currently playing digital audio content. Figure 2 shows different examples of computing platforms 103 that can act as home or business audio gateways, though these examples are by no means complete or exhaustive in their coverage of the possible options for computing platforms 103 that can act as an audio gateway.

The first example is typically the most common type of computing platform 103, a personal computer 105. The personal computer 105 is able to connect to the Internet or other computer network 102 using an internal or external network interface or modem

137. The personal computer 105 is able to cache digital audio content and other information 101 downloaded from a server 100 connected to the Internet or other computer network 102 into persistent storage 133, such as a hard drive, on the personal computer 105. Access to a wireless network or proprietary wireless communication 104 can be provided either by an external wireless network access point 106 or an internal or external wireless network interface or proprietary wireless communication interface 141. The wireless network or proprietary wireless communication 104 allows the personal computer 105 to communicate with other computing platforms 103 as well as with mobile digital audio players 115 and fixed digital audio players 116.

Another example of a computing platform 103 that can act as an audio gateway is a set-top box 107. The most common use of the set-top box 107 is to receive analog or digital audio or video 151 from sources such as cable or satellite, then decode the analog or digital audio or video 151 using an analog or digital audio and video decoder 139. The set-top box then provides this audio output 152 and video output 150 to an audio and video playback device, such as a television set 108. As an audio gateway, the set-top box 107 takes on some additional functionality. A connection from the set-top box 107 to the Internet or other computer network 102 can be provided through the same cable or satellite connection, or through a separate connection using an internal or external network interface or modem 137. The set-top box 107 is able to cache digital audio content and other information 101 downloaded from a server 100 connected by the Internet or other computer network 102 into persistent storage 133, such as a hard drive, on the set-top box 107. Access to a wireless network or proprietary wireless communication 104 can typically be provided by an internal or external wireless network interface or proprietary wireless communication interface 141. The wireless network or proprietary wireless communication 104 allows the set-top box 107 to communicate with other computing platforms 103 as well as with mobile digital audio players 115 and fixed digital audio players 116.

A final example of a computing platform 103 that can act as an audio gateway is a stand-alone audio gateway 109. The stand-alone audio gateway 109 acts as a fixed function device, whose main purpose is to be an audio gateway. The fixed function nature of the stand-alone audio gateway 109 is unlike the personal computer 105 and the set-top box 107 examples, which provide other functionality beyond the features required for an audio gateway. The stand-alone audio gateway 109 is able to connect to the Internet or other computer network 102 using an internal or external network interface or modem 137. The stand-alone audio gateway 109 is able to cache digital audio content and other information 101 downloaded from a server 100 connected to the Internet or other computer network 102 into persistent storage 133, such as a hard drive, on the stand-alone audio gateway 109. Access to a wireless network or proprietary wireless communication 104 can typically be provided by an internal or external wireless network interface or proprietary wireless communication interface 141. The wireless network or proprietary wireless communication 104 allows the stand-alone audio gateway 109 to communicate with other computing platforms 103 as well as with mobile digital audio players 115 and fixed digital audio players 116.

### *Local Wireless Network*

The wireless network or proprietary wireless communication 104 provides the communications transport between a computing platform 103, acting as an audio gateway, and mobile digital audio players 115 and fixed digital audio players 116. Examples of computing platforms 103 acting as an audio gateway include personal computers 105, set-top boxes 107, and stand-alone audio gateways 109. Wireless communications between the computing platform 103 and mobile digital audio players 115 and fixed digital audio players 116, illustrated in figure 3, can be done using industry standard wireless communications and networking technology, such as Bluetooth, HomeRF, and IEEE 802.11. In addition, with respect to this invention, a proprietary wireless communications technology may also be used for wireless communications. Use of the wireless network or proprietary wireless communication 104 by computing platforms 103, mobile digital audio players 115, and fixed digital audio players 116 may be handled as an internal or external peripheral in the form of a wireless network interface or proprietary wireless communication interface 141. The wireless network or proprietary wireless communication 104 may also require an external wireless access point 106 to handle or facilitate wireless communications and to act as a bridge between the wireless network and wired networking connections, such as may be used by a personal computer 105.

#### ***Example Configurations***

Figures 4 through 6 present specific examples that demonstrate various aspects of the invention. These examples are by no means the only possible configurations that support the invention and do not necessarily cover all aspects of the invention.

#### ***Personal Computer and Digital Audio Players Configuration***

The first example configuration, shown in figure 4, uses a personal computer 105 as the audio gateway. The personal computer 105 connects to the Internet or other computer network 102 using a network interface or modem 137. The personal computer 105 can download digital audio content and other information 101 from a server 100 connected to the Internet or other computer network 102. The digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, can then be stored in persistent storage 133, such as a hard drive, on the personal computer 105. The user can also create new playlists using the personal computer 105. In this example, a wireless access point 106 is used to handle the wireless network or proprietary wireless communication 104. The wireless network or proprietary wireless communication 104 is used by the personal computer 105, acting as the audio gateway, to communicate with mobile digital audio players 115 and fixed digital audio players 116. The personal computer 105, using the wireless network or proprietary wireless communication 104, is able to, either automatically or at user request, pass the digital audio content and other information 101, including new playlists, to mobile digital audio players 115 and fixed digital audio players 116. If a fixed digital audio player 116, such as a stand-alone player 112 or a rack player 113 that connects to a stereo 114, happens to be turned off at the time, then the personal computer 105 is able to automatically detect the next time the fixed digital audio player 116 is turned on. When the personal computer 105 detects that a fixed digital audio player 116 has just turned on, then the personal computer 105 can

pass the digital audio content and other information 101 to the fixed digital audio player 116 at that time. Mobile digital audio players 115, such as automotive players 110 and handheld players 111, are typically out of range of the wireless network or proprietary wireless communication 104 during normal use. When a mobile digital audio player 115 comes into range of the wireless network or proprietary wireless communication 104, the personal computer 105, acting as an audio gateway, can automatically detect the mobile digital audio player 115 and pass the digital audio content and other information 101 at that time. In addition, the personal computer 105 can, either automatically or upon user request, determine the current playlist and current position within the playlist on a particular mobile digital audio player 115 or fixed digital audio player 116. Then the personal computer 105 can propagate this playlist information to any other mobile digital audio players 115 and fixed digital audio players 116 that are on and in range. This allows a user to move from one mobile digital audio player 115 or fixed digital audio player 116 to another mobile digital audio player 115 or fixed digital audio player 116 and automatically be able to continue the same music and playlist in a seamless manner.

#### ***Set-top Box and Digital Audio Players Configuration***

Another example configuration, shown in figure 5, uses a set-top box 107 as the audio gateway. The set-top box 107 can connect to the Internet or other computer network 102 either through the same cable or satellite connection that provides the analog or digital audio or video 151 that is passed to an audio or video playback device, such as a television set 108, or through an internal or external network interface or modem 137. The set-top box 107 can download digital audio content and other information 101 from a server 100 connected to the Internet or other computer network 102. The digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, can then be stored in persistent storage 133, such as a hard drive or flash memory, on the set-top box 107. In this example, a wireless network interface or proprietary wireless communication interface 141 is used to handle the wireless network or proprietary wireless communication 104. The set-top box 107, acting as the audio gateway, uses the wireless network or proprietary wireless communication 104 to communicate with mobile digital audio players 115 and fixed digital audio players 116. The set-top box 107, using the wireless network or proprietary wireless communication 104, is able to, either automatically or at user request, pass the digital audio content and other information 101 to mobile digital audio players 115 and fixed digital audio players 116. If a fixed digital audio player 116, such as a stand-alone player 112 or a rack player 113 that connects to a stereo 114, happens to be turned off at the time, then the set-top box 107 is able to automatically detect the next time the fixed digital audio player 116 is turned on. When the set-top box 107 detects that a fixed digital audio player 116 has just turned on, then the set-top box 107 can pass the digital audio content and other information 101 to the fixed digital audio player 116 at that time. Mobile digital audio players 115, such as automotive players 110 and handheld players 111, are typically out of range of the wireless network or proprietary wireless communication 104 during normal use. When a mobile digital audio player 115 comes into range of the wireless network or proprietary wireless communication 104, the set-top box 107, acting as an audio gateway, can automatically detect the mobile digital audio player 115 and pass the



digital audio content and other information 101 at that time. In addition, the set-top box 107 can determine, either automatically or upon user request, the current playlist and current position within the playlist on a particular mobile digital audio player 115 or fixed digital audio player 116. Then the set-top box 107 can propagate this playlist information to any other mobile digital audio players 115 and fixed digital audio players 116 that are on and in range. This allows a user to move from one mobile digital audio player 115 or fixed digital audio player 116 to another mobile digital audio player 115 or fixed digital audio player 116 and automatically be able to continue the same music and playlist in a seamless manner.

#### ***Stand-alone Gateway and Digital Audio Players Configuration***

A final example configuration, shown in figure 6, uses a stand-alone audio gateway 109 as the audio gateway. The stand-alone audio gateway 109 connects to the Internet or other computer network 102 using a network interface or modem 137. The stand-alone audio gateway 109 can download digital audio content and other information 101 from a server 100 connected to the Internet or other computer network 102. The digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, can then be stored in persistent storage 133, such as a hard drive or flash memory, on the stand-alone audio gateway 109. In this example, a wireless network interface or proprietary wireless communication interface 141 is used to handle the wireless network or proprietary wireless communication 104. The wireless network or proprietary wireless communication 104 is used by the stand-alone audio gateway 109 to communicate with mobile digital audio players 115 and fixed digital audio players 116. The stand-alone audio gateway 109, using the wireless network or proprietary wireless communication 104, is able to, either automatically or at user request, pass the digital audio content and other information 101 to mobile digital audio players 115 and fixed digital audio players 116. If a fixed digital audio player 116, such as a stand-alone player 112 or a rack player 113 that connects to a stereo 114, happens to be turned off at the time, then the stand-alone audio gateway 109 is able to automatically detect the next time the fixed digital audio player 116 is turned on. When the stand-alone audio gateway 109 detects that a fixed digital audio player 116 has just turned on, then the stand-alone audio gateway 109 can pass the digital audio content and other information 101 to the fixed digital audio player 116 at that time. Mobile digital audio players 115, such as automotive players 110 and handheld players 111, are typically out of range of the wireless network or proprietary wireless communication 104 during normal use. When a mobile digital audio player 115 comes into range of the wireless network or proprietary wireless communication 104, the stand-alone audio gateway 109 can automatically detect the mobile digital audio player 115 and pass the digital audio content and other information 101 at that time. In addition, the stand-alone audio gateway 109 can, either automatically or upon user request, determine the current playlist and current position within the playlist on a particular mobile digital audio player 115 or fixed digital audio player 116. Then the stand-alone audio gateway 109 can propagate this playlist information to any other mobile digital audio players 115 and fixed digital audio players 116 that are on and in range. This allows a user to move from one mobile digital audio player 115 or fixed digital audio player 116 to another mobile digital audio player 115 or

fixed digital audio player 116 and automatically be able to continue the same music and playlist in a seamless manner.

#### ***Communication Between Digital Audio Players***

Another aspect of this invention, shown in figure 7, is the ability for mobile digital audio players 115 and fixed digital audio players 116 to wirelessly communicate amongst themselves. The mobile digital audio players 115 and fixed digital audio players 116, use the same wireless network or proprietary wireless communication 104 that is used to wirelessly communicate with the computing platform 103, to communicate with each other. This wireless communication between the mobile digital audio players 115 and fixed digital audio players 116 themselves is handled by an internal or external wireless network interface or proprietary wireless communication interface 141 in each of the mobile digital audio players 115 and fixed digital audio players 116. This wireless communication includes directly passing digital audio content and other information 101, including playlists from one mobile digital audio players 115 or fixed digital audio players 116 to another. Examples of mobile digital audio players 115 include automotive players 110 and handheld players 111, and examples of fixed digital audio players include stand-alone players 112 and rack players 113 that connect to a stereo 114.

#### ***Computing Platform***

Figure 8 shows the typical system architecture of a computing platform 103, which can encompass anything from general-purpose devices, such as personal computers 105, to open fixed function devices, such as set-top boxes 107 or stand-alone audio gateways 109, among others. In general, the computing platform 103 has a main processor 130 for executing various software components. The various software components are typically stored in read only memory, or ROM, or flash memory 136, or local storage 132. Local storage 132 can consist of persistent storage 133, such as hard drives or flash memory, or removable storage 134 such as floppy drives, CD-ROM drives, or DVD drives. The software components are executed by the main processor 130 directly from their storage location or are loaded into random access memory or RAM 135, to be executed from RAM 135 by the main processor 130. Local storage 132 can also be used to cache digital audio content and other information 101. The computing platform 103 uses a network interface or modem 137 to access servers 100 on the Internet or other computer network 102, in order to download digital audio content or other information 101. The network interface or modem 137 is connected internally or externally to the computing platform 103 using a system bus or peripheral bus 131. The system bus and peripheral buses 131 are provided for connecting internal and external devices to the computing platform 103 in a standard manner. Typical system and peripheral buses 131 include Universal Serial Bus, commonly referred to as USB, IEEE 1394, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI. The computing platform 103 also supports connection through a user input interface 142 to external or integrated user input devices 153, such as keyboards and mice. For output to the user, the computing platform 103 may contain a display controller 138, which stores graphical data such as windows, bitmaps and text. The display controller 138 outputs the graphical data in a video output 150 format that is typically displayed to the user on a video monitor,

television 108, or LCD panel. In addition to video output 150, the computing platform 103 can provide audio output 152, which is handled by audio playback hardware 140. For a computing platform 103 that is acting as a set-top box 107, the computing platform 103 will likely also contain an analog or digital audio and video decoder 139. The analog or digital audio and video decoder 139 decodes the analog or digital audio or video 151 from sources such as cable or satellite, and passes the audio output 152 and video output 150 to an audio and video playback device, such as a television set 108. For wireless communication with other computing platforms 103, mobile digital audio players 115, and fixed digital audio players 116 on a wireless network or proprietary wireless communication 104, the computing platform 103 uses an internal or external wireless network interface or proprietary wireless communication interface 141. It should be noted that a computing platform 103 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or features not listed.

#### *Handheld Player*

There are many different types of mobile digital audio players 115. Figure 9 demonstrates the general architecture for a handheld player 111. In general, a handheld player 111 has a processor 155 that is responsible for executing various software and firmware components. The various software and firmware components are typically stored in read only memory, or ROM, or flash memory 158 or in player storage 156, such as a hard drive, flash memory, or removable media. The software and firmware components are executed by the processor 155 directly from their storage location or are loaded into random access memory or RAM 157 to be executed from RAM 157 by the processor 155. Player storage 156 can also be used for storing digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, for later playback and presentation to the user. Typically, the digital audio content 101 is in some encoded format. The audio decoder 162 decodes the digital audio content 101 and passes it to the audio digital to analog converter 163, or DAC. The audio DAC 163 converts the decoded audio to analog and then provides audio output 166 from the handheld player 111. The audio output 166 of a handheld player 111 typically is used to drive headphones. Communication using a wireless network or proprietary wireless communication 104 by the handheld player 111 with computing platforms 103, other mobile digital audio players 115, and fixed digital audio players 116 is done using an internal or external wireless network interface or proprietary wireless communication interface 141. For input from the user, the handheld player 111 contains user inputs 165, such as buttons or a touch screen. The user input interface 164 handles the actual interface with the user inputs 165, while interpretation of these inputs are typically handled by software and firmware running on the processor 155. For output to the user, the handheld player 111 may contain a display controller 160, which can provide text and possibly graphical output to the user on an LCD display 161. Tying of the functional components and processor 155 together is typically done using a system bus and peripheral buses 159. Examples of system and peripheral buses 159 include Universal Serial Bus, commonly referred to as USB, IEEE 1394, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI. It should be noted that

some of the functional blocks described might encompass multiple physical components. As well, multiple functional blocks may be contained in a single physical component. It should also be noted that a handheld player 111 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or features not listed.

### *Automotive Player*

Another type of mobile digital audio player 115 is the automotive player 110, whose general architecture is shown in figure 10. In general, an automotive player 110 has a processor 155 that is responsible for executing various software and firmware components. The various software and firmware components are typically stored in read only memory, or ROM, or flash memory 158 or in player storage 156, such as a hard drive, flash memory, or removable media. The software and firmware components are executed by the processor 155 directly from their storage location or are loaded into random access memory or RAM 157 to be executed from RAM 157 by the processor 155. Player storage 156 can also be used for storing digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, for later playback and presentation to the user. Typically, the digital audio content 101 is in some encoded format. The audio decoder 162 decodes the digital audio content 101 and passes it to the audio digital to analog converter 163, or DAC. The audio DAC 163 converts the decoded audio to analog and then provides audio output 167 from the automotive player 110. The audio output 167 of an automotive player 110 typically feeds some sort of car audio amplifier, which then drives the car speakers. Communication using a wireless network or proprietary wireless communication 104 by the automotive player 110 with computing platforms 103, other mobile digital audio players 115, and fixed digital audio players 116 is done using an internal or external wireless network interface or proprietary wireless communication interface 141. For input from the user, the automotive player 110 contains user inputs 165, such as buttons or a touch screen. The user input interface 164 handles the actual interface with the user inputs 165, while interpretation of these inputs are typically handled by software and firmware running on the processor 155. In addition, an automotive player 110 may support voice commands for user input. If voice commands are supported, a microphone 174 is used to feed analog audio to the audio analog to digital converter 173, which converts the analog audio to digital. Then, the audio capture hardware 172 and the processor 155 will interpret the voice commands from the user. For output to the user, the automotive player 110 may contain a display controller 160, which can provide text and possibly graphical output to the user on an LCD display 161. Tying of the functional components and processor 155 together is typically done using a system bus and peripheral buses 159. Examples of system and peripheral buses 159 include Universal Serial Bus, commonly referred to as USB, IEEE 1394, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI. It should be noted that some of the functional blocks described might encompass multiple physical components. As well, multiple functional blocks may be contained in a single physical component. It should also be noted that an automotive player 110 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or

features not listed.

### ***Rack Player***

There are many different types of fixed digital audio players 116. Figure 11 demonstrates the general architecture for a rack player 113. In general, a rack player 113 has a processor 155 that is responsible for executing various software and firmware components. The various software and firmware components are typically stored in read only memory, or ROM, or flash memory 158 or in player storage 156, such as a hard drive, flash memory, or removable media. The software and firmware components are executed by the processor 155 directly from their storage location or are loaded into random access memory or RAM 157 to be executed from RAM 157 by the processor 155. Player storage 156 can also be used for storing digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, for later playback and presentation to the user. Typically, the digital audio content 101 is in some encoded format. The audio decoder 162 decodes the digital audio content 101 and passes it to the audio digital to analog converter 163, or DAC. The audio DAC 163 converts the decoded audio to analog and then provides audio output 167 from the rack player 113. The audio output 167 of a rack player 113 typically is passed to a stereo system 114. Communication using a wireless network or proprietary wireless communication 104 by the rack player 113 with computing platforms 103, mobile digital audio players 115, and other fixed digital audio players 116 is done using an internal or external wireless network interface or proprietary wireless communication interface 141. For input from the user, the rack player 113 contains user inputs 165, such as buttons or a touch screen. The rack player 113 may also receive infrared input 168 from a remote control. The user input interface 164 handles the actual interface with the user inputs 165 and the infrared input 168, while interpretation of these inputs are typically handled by software and firmware running on the processor 155. For output to the user, the rack player 113 may contain a display controller 160, which can provide text and possibly graphical output to the user on an LCD display 161. Tying of the functional components and processor 155 together is typically done using a system bus and peripheral buses 159. Examples of system and peripheral buses 159 include Universal Serial Bus, commonly referred to as USB, IEEE 1394, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI. It should be noted that some of the functional blocks described might encompass multiple physical components. As well, multiple functional blocks may be contained in a single physical component. It should also be noted that a rack player 113 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or features not listed.

### ***Stand-alone Player***

Another type of fixed digital audio players 116 is the stand-alone player 112, whose general architecture is shown in figure 12. In general, a stand-alone player 112 has a processor 155 that is responsible for executing various software and firmware components. The various software and firmware components are typically stored in read only memory, or ROM, or flash memory 158 or in player storage 156, such as a hard

drive, flash memory, or removable media. The software and firmware components are executed by the processor 155 directly from their storage location or are loaded into random access memory or RAM 157 to be executed from RAM 157 by the processor 155. Player storage 156 can also be used for storing digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, for later playback and presentation to the user. Typically, the digital audio content 101 is in some encoded format. The audio decoder 162 decodes the digital audio content 101 and passes it to the audio digital to analog converter 163, or DAC. The audio DAC 163 converts the decoded audio to analog. The analog audio from a stand-alone player 112 typically directly drives speakers 170 attached to the stand-alone player 112. Communication using a wireless network or proprietary wireless communication 104 by the stand-alone player 112 with computing platforms 103, mobile digital audio players 115, and other fixed digital audio players 116 is done using an internal or external wireless network interface or proprietary wireless communication interface 141. For input from the user, the stand-alone player 112 contains user inputs 165, such as buttons or a touch screen. The stand-alone player 112 may also receive infrared input 168 from a remote control. The user input interface 164 handles the actual interface with the user inputs 165 and the infrared input 168, while interpretation of these inputs are typically handled by software and firmware running on the processor 155. For output to the user, the stand-alone player 112 may contain a display controller 160, which can provide text and possibly graphical output to the user on an LCD display 161. Tying of the functional components and processor 155 together is typically done using a system bus and peripheral buses 159. Examples of system and peripheral buses 159 include Universal Serial Bus, commonly referred to as USB, IEEE 1394, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI. It should be noted that some of the functional blocks described might encompass multiple physical components. As well, multiple functional blocks may be contained in a single physical component. It should also be noted that a stand-alone player 112 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or features not listed.

#### ***Audio Gateway Software***

Figures 13 to 17 provide flow diagrams for the key software functionality that represents the software component of this invention with respect to the audio gateway. In these flow diagrams, the software is assumed to be running in a multitasking environment, with each of the flow diagrams representing a particular independently running task or process. However, it should be noted that these flow diagrams represent only one of many different ways to implement the key software functionality for the audio gateway and that many other implementations are possible, including those which do not require a multitasking environment.

#### ***Audio Gateway Message Handling Flow***

Figure 13 provides the flow diagram of the message handler for the audio gateway. In general, the message handler takes the messages received from other computing platforms 103, mobile digital audio players 115, and fixed digital audio players 116 on

the wireless network or proprietary wireless communication 104 and queues these messages for use by other processes or handles them itself, depending on the message type. In this example, the message handler is a continuously running process. Start 200 represents the beginning of the message handling process. The message handler checks if there is a message received 201. If there is a message received 201, the message handler then checks to see what type of message it is, among many possible types. If the message is a broadcast response message from a player 202, then the message handler queues the broadcast response message 203. After the message handler queues the broadcast response message 203, the message handler checks for more messages. If the message is a query response message from a player 204, then the message handler queues the query response message 205. After the message handler queues the query response message 205, the message handler checks for more messages. If the message is a poll response message from a player 206, then the message handler queues the poll response message 207. After the message handler queues the poll response message 207, the message handler checks for more messages. If the message is a playlist response message from a player 208, then the message handler queues the playlist response message 209. After the message handler queues the playlist response message 209, the message handler checks for more messages. If the message is a content response message from a player 210, then the message handler queues the content response message 211. After the message handler queues the content response message 211, the message handler checks for more messages. If the message is a content acknowledge message from a player 212, then the message handler queues the content acknowledge message 213. After the message handler queues the content acknowledge message 213, the message handler checks for more messages. Finally, if the message was none of those previously checked for, the message handler handles or queues any other messages as necessary 214.

#### ***Audio Gateway Discovery Flow***

Discovery of mobile digital audio players 115 and fixed digital audio players 116 within range of the audio gateway, on the wireless network or proprietary wireless communication 104, is an important capability with respect to this invention. Figure 14 provides the flow diagram for discovery by the audio gateway of mobile digital audio players 115 and fixed digital audio players 116. In this example, the audio gateway discovery handler is a continuously running process. Start 220 represents the beginning of the discovery handling process. In order to get a message response from mobile digital audio players 115 and fixed digital audio players 116, the discovery handler sends a broadcast for players message 221. The discovery handler then waits, with some timeout, for a broadcast response message from any players 222. The discovery handler then checks if there is a player broadcast response message in the queue 223. If there is no response, then the discovery handler broadcasts again for players. If there is a response, then the discovery handler sends a query player message to a responding player 224 to get information about the type of player that has responded. The discovery handler then waits, with some timeout, for a player query response message 225 from the player that previously responded to the broadcast. The discovery handler then checks if there is a query response message in the queue 226. If there is no response, then the discovery handler broadcasts again for players. If there is a response, then the discovery handler

checks the information returned in the query response message to see if the player is already known 227. If the player is already known 227, then the discovery handler broadcasts again for players. However, a player is unlikely to respond to a broadcast from an audio gateway when the player and audio gateway already know about each other. If the player is not already known 227, then the discovery handler adds the player to the list of players in proximity 228 of the audio gateway. Finally, the discovery handler flags the new player in proximity for playlist continuation 229 and for content synchronization 230. This allows the playlist continuation handler in the audio gateway to capture the current playlist and current selection from this new player for possible broadcast to other players. Also, this allows the content synchronization handler in the audio gateway to automatically download digital audio content and other information 101 cached on the audio gateway to the new player.

#### *Audio Gateway Dropout Detection Flow*

The flow diagram for audio gateway detection of dropout of players is shown in figure 15. The dropout detection handler in the audio gateway polls players that are known to be in proximity in order to see if any of the players has possibly gone out of range of the wireless network or proprietary wireless communication 104 or has been turned off. In this example, the dropout detection handler is a continuously running process. Start 240 represents the beginning of the dropout detection handling process. The dropout detection handler checks the list of players in proximity 241 maintained by the audio gateway. If there are players in proximity 242, then the dropout detection handler sends a poll message to the next player in proximity in the proximity list 243. This allows all the players in the list of players in proximity to be checked in a sequential manner. Then the dropout detection handler waits, with some timeout, for a poll response message from the player 244 that was sent the poll message 243. If there is no poll response message from the player in the queue 245 then the dropout detection handler checks if the player is already flagged as possibly being out of range 246 of the wireless network or proprietary wireless communication 104. If the player is not already flagged as possibly out of range 246, then the dropout detection handler flags that the player is possibly out of range 247 and checks the list of players in proximity 241 again. If the player is already flagged as possibly out of range 246, then the dropout detection handler removes the player from the list of players in proximity 248 and checks the list of players in proximity 241 again. If the player poll response message is in the queue 245, then the dropout detection handler clears the possibly out of range flag 249 for the player in the list of players in proximity. Next, the dropout detection handler checks if the player is requesting content synchronization 250, based on information passed in the poll response message from the player. If the player is requesting content synchronization 250, then the dropout detection handler flags the player for content synchronization 251 in the list of players in proximity. The content synchronization handler uses this information when deciding which players to update for digital audio content and other information 101. Once the player is flagged for content synchronization 251 or the player is not requesting content synchronization 250, then the dropout detection handler checks if the player is requesting playlist continuation 252, based on information passed in the poll response message from the player. If the player is requesting playlist continuation 252, then the dropout detection



handler flags the player for playlist continuation 253 in the list of players in proximity. The playlist continuation handler uses this information when deciding which players to update the playlist and current selection for. Once the player is flagged for playlist continuation 253 or the player is not requesting playlist continuation 252, then the dropout detection handler checks the list of players in proximity 241 again.

#### ***Audio Gateway Content Synchronization Flow***

The flow diagram for audio gateway content synchronization is shown in figure 16, with content synchronization being a key capability of the invention. The content synchronization handler in the audio gateway checks for players that need content synchronization. Content synchronization involves updating or adding digital audio content and other information 101 to a player when the audio gateway has digital audio content and other information 101 that is not contained on the player. This is typically handled automatically when the player has recently been discovered as being in proximity by the gateway discovery handler or the player directly requests content synchronization through poll response messages to the gateway. In this example, the content synchronization handler is a continuously running process. Start 260 represents the beginning of the content synchronization handling process. The content synchronization handler checks the list of players in proximity 261 maintained by the gateway. If there are players in proximity flagged for content synchronization 262, then the content synchronization handler sends a query player for content message to the player 263 that is flagged for content synchronization. Next, the content synchronization handler waits, with some timeout, for a player content response message 264. If there is no content response message in the queue 265 from the player that was sent the query player for content message 263, then the content synchronization handler clears the content synchronization flag for the player in the proximity list 266 and checks the list of players in proximity 261 again. If there is a content response message in the queue 265 from the player that was sent the query player for content message 263, then the gateway compares the digital audio content in the player with the digital audio content in the gateway 267. The player's digital audio content information is contained in the content response message sent to the gateway by the player. Next, the content synchronization handler checks if there is any content in the gateway that is not on the player 268. If the player content is properly synchronized with the gateway, then the content synchronization handler clears the content synchronization flag for the player in the proximity list 266 and checks the list of players in proximity 261 again. If there is content on the gateway that is not on the player 268, then the content synchronization handler checks if there is storage on the player for the new content 269. The available storage on the player is provided in the content response message that the player sent to the gateway. If there is not sufficient storage on the player for the new content 269, then the content synchronization handler clears the content synchronization flag for the player in the proximity list 266 and checks the list of players in proximity 261 again. If there is storage on the player for the new content 269, then the content synchronization handler sends the content data to the player 270. Next, the content synchronization handler waits, with some timeout, for the content acknowledge message from the player 271. If there is no content acknowledge message in the queue 272, then the content synchronization handler

clears the content synchronization flag for the player in the proximity list 266 and checks the list of players in proximity 261 again. If there is a content acknowledge message in the queue 272 from the player, then the content synchronization handler checks to see, from the compare of content in the player with content in the gateway 267, if there is more content to send to the player 273. If there is more content to send to the player 273 then the content synchronization handler checks again if there is storage on the player for the new content 269, and so on until there is no more content to pass from the gateway to the player. If there is no more content to send to the player 273, then the content synchronization handler clears the content synchronization flag for the player in the proximity list 266 and checks the list of players in proximity 261 again.

#### ***Audio Gateway Playlist Continuation Flow***

The flow diagram for audio gateway playlist continuation is shown in figure 17, with playlist continuation being a key capability of the invention. The playlist continuation handler in the audio gateway checks for propagation of the playlist and current playlist selection from one mobile digital audio player 115 or fixed digital audio player 116 to all other mobile digital audio players 115 and fixed digital audio players 116 in proximity. Playlist continuation involves seamless continuation of playback of digital audio content 101 from a particular playlist as a user moves from one mobile digital audio player 115 or fixed digital audio player 116 to another. This is typically handled automatically when the gateway discovery handler discovers a player as being in proximity, where the player is currently playing digital audio content 101. The player itself may also directly request playlist continuation through poll response messages to the gateway. In this example, the playlist continuation handler is a continuously running process. Start 280 represents the beginning of the playlist continuation handling process. The playlist continuation handler checks the list of players in proximity 281 maintained by the gateway. If there are players in proximity flagged for playlist continuation 282, then the playlist continuation handler sends a query player for playlist message to the player 283 that is flagged for playlist continuation. Next, the playlist continuation handler waits, with some timeout, for a player playlist response message 284. If there is no playlist response message in the queue 285 from the player that was sent the query player for playlist message 283, then the playlist continuation handler clears the playlist continuation flag for the player in the proximity list 286 and checks the list of players in proximity 281 again. If there is a playlist response message in the queue 285 from the player that was sent the query player for playlist message 283, then the gateway checks the playlist response message to see if the playlist and current position within the playlist, both of which are contained in the playlist response message, are valid 287. If the playlist and current position are not valid 287, then the playlist continuation handler clears the playlist continuation flag for the player. If the playlist and current position in the playlist are valid 287, then the playlist continuation handler checks the list of players in proximity 288. If there are any other players in proximity 289, then the playlist continuation handler sends a broadcast playlist and current position message to all other players in proximity 290. After the playlist continuation handler sends a broadcast playlist and current position message to all other players in proximity 290 or if there are no other players in proximity 289, then the playlist continuation handler clears the playlist continuation flag for the player in the

proximity list 286 and checks the list of players in proximity 281 again.

### ***Player Software***

Figures 18 to 24 provide flow diagrams for the key software functionality that represents the software component of this invention with respect to the player. In these flow diagrams, the software is assumed to be running in a multitasking environment, with each of the flow diagrams representing a particular independently running task or process. However, it should be noted that these flow diagrams represent only one of many different ways to implement the key software functionality for the player and that many other implementations are possible, including those which do not require a multitasking environment.

### ***Player Message Handling Flow***

Figure 18 provides the flow diagram of the message handler for a player. In general, the message handler takes the messages received from computing platforms 103 acting as audio gateways and from other mobile digital audio players 115 and fixed digital audio players 116, on a wireless network or proprietary wireless communication 104, and queues these messages for use by other processes or handles them itself, depending on the message type. In this example, the message handler is a continuously running process. Start 300 represents the beginning of the message handling process. The message handler checks if there is a message received 301. If there is a message received 301, the message handler then checks to see what type of message it is, among many possible types. If the message is a broadcast for players message from a gateway 302, then the message handler queues the broadcast for players message 303. After the message handler queues the broadcast for players message 303, the message handler checks for more messages. If the message is a query player message from a gateway 304, then the message handler queues the query player message 305. After the message handler queues the query player message 305, the message handler checks for more messages. If the message is a poll message from a gateway 306, then the message handler queues the poll message 307. After the message handler queues the poll message 307, the message handler checks for more messages. If the message is a query player for content message from a gateway 308, then the message handler queues the query player for content message 309. After the message handler queues the query player for content message 309, the message handler checks for more messages. If the message is content data from a gateway 310, then the message handler stores the content in local player storage 311. The message handler also sends a content acknowledge message to the gateway 312. After the message handler sends a content acknowledge message to the gateway 312, the message handler checks for more messages. If the message is a query player for playlist message from a gateway 313, then the message handler queues the query player for playlist message 314. After the message handler queues the query player for playlist message 314, the message handler checks for more messages. If the message is a broadcast playlist message from a gateway 315, then the message handler queues the broadcast playlist message 316. After the message handler queues the broadcast playlist message 316, the message handler checks for more messages. Finally, if the message was none of those previously checked for, the message handler handles or queues any other messages as necessary 317 and then the

message handler checks for more messages.

### ***Player Discovery Flow***

Discovery by the audio gateway of mobile digital audio players 115 and fixed digital audio players 116 is an important capability with respect to this invention. Figures 19 and 20 provide the flow diagrams for discovery responses by the player when the player detects discovery attempts by an audio gateway. In this example, the player discovery broadcast response handler and the player discovery query response handler are continuously running processes. Start 320 represents the beginning of the discovery broadcast response handling process. The discovery broadcast response handler first checks for a broadcast for players message in the queue 321 from a gateway. If there is a broadcast for players message in the queue 322, then the discovery broadcast response handler checks if the gateway is already in proximity of the player 323. The discovery broadcast response handler is able to get information about the gateway from the broadcast for players message received from the gateway and can compare that information with information saved by the discovery query response handler for any gateway in proximity. If the gateway is not already in proximity 323, then the discovery broadcast response handler sends a broadcast acknowledge message to the gateway 324. After the discovery broadcast response handler sends the broadcast acknowledge message to the gateway 324, or if the gateway is already in proximity 323, or if there is no broadcast for players message in the queue 322, then the discovery broadcast response handler checks for a broadcast for players message in the queue 321 again.

Start 330 represents the beginning of the discovery query response handling process. The discovery query response handler first checks for query player messages from a gateway in the queue 331. If there is a query player message in the queue 332, then the discovery query response handler sends a query response message to the gateway 333 that sent the query player message. Then the discovery query response handler saves that the gateway is in proximity 334 from information obtained from the query player message from the gateway. After the discovery query response handler saves that the gateway is in proximity 334 or if there is no query player message in the queue 332, then the discovery query response handler checks for a query player message from a gateway in the queue 331 again.

### ***Player Dropout Detection Flow***

The flow diagram for player dropout detection of an audio gateway is shown in figure 21. The dropout detection handler in the player watches for poll messages from an audio gateway in order to see if the player has gone out of range of the gateway. In this example, the player dropout detection handler is a continuously running process. Start 340 represents the beginning of the player dropout detection handling process. The player dropout detection handler checks if the player is in proximity of a gateway 341. The player discovery query response handler, shown in figure 20, saves information about a gateway that is in proximity. If the player is not in proximity of a gateway 341, then the player discovery query response handler just continues to check if the player is in proximity of a gateway 341. If the player is in proximity of a gateway 341, then the

player discovery query response handler waits, with some timeout, for a poll response message from the gateway 342 that is in proximity. Obviously this timeout period is significantly more than the polling period used by the gateway. If there is not a poll message in the queue 343 from the gateway that is in proximity, then the player discovery query response handler checks if the gateway is already flagged as possibly out of range 344. If the gateway is not already flagged as possibly out of range 344, then the player discovery query response handler flags that the gateway is possibly out of range 345 and continues to check if the player is in proximity of a gateway 341. If the gateway is already flagged as possibly out of range 344, then the player discovery query response handler removes the gateway as being in proximity 346 and then continues to check if the player is in proximity of a gateway 341. If there is a poll message in the queue 343 from the gateway that is in proximity, then the player discovery query response handler checks if the user requested content synchronization of the player 347. If the user did request content synchronization of the player 347, then the player discovery query response handler flags a content synchronization request in the poll response message 348 to the gateway in proximity. If the user did not request content synchronization of the player 347, then the player discovery query response handler skips flagging of content synchronization in the poll response message 348. Next, the player discovery query response handler checks if the user requested playlist continuation for the player 349. If the user did request playlist continuation for the player 349, then the player discovery query response handler flags a playlist continuation request in the poll response message 350 to the gateway in proximity. If the user did not request playlist continuation for the player 349, then the player discovery query response handler skips flagging of playlist continuation in the poll response message 350. Next, the player discovery query response handler sends the poll response message to the gateway 351 that is in proximity and sent the poll message. Next, the player discovery query response handler clears the gateway possibly out of range flag 352 if it was set for the gateway in proximity. Then the player discovery query response handler continues to check if the player is in proximity of a gateway 341.

#### ***Player Content Synchronization Flow***

The flow diagram for player content synchronization response is shown in figure 22, with content synchronization being a key capability of the invention. The content synchronization response handler in the player responds to content queries from a gateway. In this example, the content synchronization response handler is a continuously running process. Start 360 represents the beginning of the content synchronization response handling process. The content synchronization response handler checks for a query player content message in the queue 361 from a gateway. If there is a query player content message in the queue 362, then the content synchronization response handler builds a content response message by first getting a list of all the digital audio content on the player 363. Next, the content synchronization response handler determines the amount of available storage space on the player 364 for additional digital audio content. Finally, the content synchronization response handler sends a player content response message 365 to the gateway that sent the query player content message. The player content response message contains the list of all the digital audio content on the player as

well as the amount of available space on the player. Once the content synchronization response handler sends a player content response message 365 to the gateway that sent the query player content message or there is no query player content message in the queue 362, then the content synchronization response handler checks for a query player content message in the queue 361 again.

### ***Player Playlist Continuation Flow***

Figures 23 and 24 provide the flow diagrams for playlist continuation response and playlist continuation updating by the player when the player detects playlist continuation query and updating attempts by an audio gateway. Playlist continuation is a key capability of the invention. In this example, the player playlist response handler and the player playlist update handler are continuously running processes. Start 380 represents the beginning of the playlist response handling process. First, the playlist response handler checks for a query player for playlist message in the queue 381 from a gateway in proximity. If there is a query player for playlist message in the queue 382, then the playlist response handler gets the current playlist and current position within the playlist 383 and puts this information in a playlist response message. Next, the playlist response handler sends the playlist response message to the gateway 384 that sent the query player for playlist message. After the playlist response handler sends the playlist response message to the gateway 384 or there is not a query player for playlist message in the queue 382, then the playlist response handler checks for a query player for playlists message 381 in the queue again.

Start 390 represents the beginning of the playlist update handling process. First the playlist update handler checks for a broadcast playlist message 391 in the queue from a gateway in proximity. If there is not a broadcast playlist message in the queue 392, then the playlist update handler just checks for a broadcast playlist message 391 in the queue again. If there is a broadcast playlist message in the queue 392, then the playlist update handler checks if the playlist already exists on the player 393. The playlist information is found in the broadcast playlist message. If the playlist already exists on the player 393, then the playlist update handler activates the playlist and sets the current position within the playlist 394 on the player. The current position within the playlist is found in the broadcast playlist message. Then the playlist update handler checks for a broadcast playlist message 391 in the queue again. If the playlist does not already exist on the player 393, then the playlist update handler saves the new playlist on the player 395. Next, the playlist update handler checks if the player is currently playing 396. If the player is not currently playing 396, then the playlist update handler sets the new playlist as the current playlist 397 and sets the current position within the playlist 394. If the player is currently playing 396, then the playlist update handler notifies the user that a new playlist is available 398. This allows the user to decide to play the new playlist or continue with a current playlist. Next, the playlist update handler checks for a broadcast playlist message 391 in the queue again.

## CLAIMS

[To be added later]

Attorney Docket No. 11748/17

"CERTIFICATE OF MAILING BY "EXPRESS MAIL"  
"Express Mail" Mailing Label Number

\_\_\_\_\_  
Date of Deposit

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" Service under 37 CFR §1.10 on the date indicated above and is addressed to the Commissioner of Patents and Trademarks, Washington, D.C. 20231.

\_\_\_\_\_  
(Typed or printed name of person mailing)

\_\_\_\_\_  
(Signature of person mailing)

**PROXIMITY SYNCHRONIZATION OF AUDIO CONTENT  
AMONG MULTIPLE PLAYBACK AND STORAGE DEVICES**

**CROSS-REFERENCE TO RELATED APPLICATIONS**

This application is related to the following commonly-owned co-pending patent applications: "Proximity Synchronizing Audio Gateway Device," by Jeffrey J. Spurgat and Hoyet H. Andrews III, Serial No. \_\_\_\_\_, Attorney Docket No. 11748/19; and "Proximity Synchronizing and Audio Playback Device," by Jeffrey J. Spurgat and Hoyet H. Andrews III, Serial No. \_\_\_\_\_, Attorney Docket No. 11748/18, both filed on even date.

**BACKGROUND OF THE INVENTION**

**1. Field of the Invention**

The present invention relates to a wireless communication system and more particularly to a wireless communication system for digital audio players for providing increased functionality including communication, interaction and synchronization between a computing platform and various digital audio players as well as communication among the digital audio players themselves.

**2. Description of the Prior Art**

A multitude of different devices for digital audio playback are known. Handheld or portable audio players, mobile as well as and fixed audio players are known. Examples of such handheld audio players are compact disc (CD) players and MP3 players. Such mobile audio players include audio players, such as CD players, mounted in vehicles. Such mobile audio



Attorney Docket No. 11748/17

players are known to be mounted either in-dash in the vehicle or in the case of conversion vans and recreational vehicles in ceiling of the vehicle. Examples of fixed digital audio playback devices include stand-alone players and rack players that are adapted to connect to a home stereo system and to a source of an AC. *such as boom boxes,*

*on AC power*  
Digital audio content from the Internet is known to be downloaded onto storage devices, such as CDs, by way of <sup>a</sup>personal computer. Such Internet-based digital audio content has also been downloaded onto ~~X~~portable MP3 audio players. Although such systems allow ~~X~~selected digital audio content to be played when desired by the user, such systems only allow rather limited functionality. As such, various functions, such as interaction, communication and synchronizing the digital content on a plurality of digital audio players must be done manually. Thus, there is a need for system for providing increased functionality of various digital audio players.

### SUMMARY OF THE INVENTION

The present invention relates to a wireless communication system and in particular to a wireless communication system for digital audio players that provides for increased functionality, such as communication, interaction and synchronization between a computing platform and various mobile, portable or fixed audio digital players as well as providing a communication link between the various digital audio players themselves. The computing platform may act, for example, through a wireless communication platform, to control the digital audio players; to act as a cache of digital audio data for the digital audio players; as well as provide a gateway to the Internet to enable the digital audio players to access additional digital audio content and other information. The computing platform may also be used to automatically update digital audio content on the digital audio players; synchronize digital audio content and playlist between digital audio players; and automatically continue a particular playlist as the user moves <sup>from</sup> to one audio digital player to another. *141 global search wireless*

### DESCRIPTION OF THE DRAWINGS

These and other advantages of the present invention will be readily understood with reference to the following specification and attached drawings wherein:

## Attorney Docket No. 11748/17

FIG. 1 is a block diagram of a digital audio communication system in accordance with the present invention.

FIG. 2 is a block diagram of a digital audio gateway in accordance with the present invention.

FIG. 3 is a block diagram of a wireless communication network which includes various digital audio players in accordance with the present invention.

FIG. 4 is a block diagram of a system which utilizes a personal computing platform for communicating with a plurality of audio players.

FIG. 5 is an alternate embodiment of the invention with illustrates the use of a television set top box as a communication link for communicating with a plurality of digital audio players in accordance with an alternate embodiment of the invention.

FIG. 6 is a block diagram of an alternate embodiment of the invention which illustrates a communication system between a number of digital audio players and stand-alone audio gateway.

FIG. 7 is a block diagram of a communication network between various digital audio players in accordance with another aspect of the present invention.

FIG. 8 is a block diagram of the computing platform in accordance with the present invention.

FIG. 9 is a block diagram of a stand-alone audio gateway in accordance with the present invention.

FIG. 10 is a block diagram of a mobile digital audio player in accordance with the present invention.

✓ FIG. 11 is a block diagram of a fixed digital audio player in accordance with the present invention.

FIG. 12 is a block diagram of a handheld or portable digital audio player in accordance with the present invention.

FIG. 13 is a block diagram of an automotive digital audio player in accordance with the present invention.

FIG. 14 is a block diagram of a rack player in accordance with the present invention.

FIG. 15 is a block diagram of a stand-alone digital audio player in accordance with the present invention.

**Attorney Docket No. 11748/17**

FIG. 16 is a flow diagram of the audio gateway message handling in accordance with the present invention.

FIG. 17 is a flow diagram of the audio gateway discovery in accordance with the present invention.

FIG. 18 is a flow diagram of the audio gateway drop-out detection in accordance with the present invention.

FIG. 19 is a flow diagram of the audio gateway content synchronization in accordance with the present invention.

FIG. 20 is a flow diagram of the audio gateway playlist continuation in accordance with the present invention.

FIG. 21 is a flow diagram of the player message handling in accordance with the present invention.

FIGS. 22 and 23 are flow diagrams of the player discovery in accordance with the present invention.

FIG. 24 is a flow diagram of the player drop-out detection with the present invention.

FIG. 25 is a flow diagram of the player content synchronization in accordance with the present invention.

FIGS. 26 and 27 are block diagrams of the player playlist continuation feature in accordance with the present invention.

**DETAILED DESCRIPTION**

The present invention is adapted to provide additional functionality of digital audio players. For example, in one embodiment, ~~as illustrated in FIG. 1~~, a computing platform 103, for example, a personal computer, is used as a gateway to enable various digital audio players 115 and 116 etc. to be connected to the Internet or other computer network 102. In this embodiment, the computing platform 103 may be configured to access one or more servers 100 on the Internet or other computer network 102 that contain digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, among other things. Though the computing platform 103 can act as a digital audio player by itself, in this embodiment of the invention, the computing platform <sup>103</sup> acts as an audio gateway for various digital audio players 115 <sup>and 116, and</sup> and can additionally provide caching of the digital audio content and

other information 101 for the digital audio players 115 and 116 etc from the servers 100 that are connected to the computing platform 103 through the Internet or other computer network 102. Using a wireless communication platform 104, the computing platform 103 is adapted to communicate with various digital audio players, such as one or more mobile digital audio players 115 and fixed digital audio players 116 that are within range of the wireless communication platform 104 forming a local wireless network as generally illustrated in FIG. 3.

Various devices are contemplated for use as audio gateways, for example, as shown in FIG. 2. In one embodiment, a personal computer 105 coupled to an internal or external wireless communication platform 104 defines an access point 106, which may be used as an audio gateway. Alternatively, a set top box 107 with a wireless communication platform 104 coupled to a conventional TV 108 may be used as an audio gateway. A stand alone gateway 109 may also be formed from a wireless communication platform 104. Other embodiments of an audio gateway are contemplated. For example, any device with a wireless communication platform, either public or private may be used.

In another embodiment of the invention, the computing platform 103 may be configured to automatically synchronize, or upon request, copy, add or remove digital audio content and other information 101, such as playlists, on mobile digital audio players 115 and fixed digital audio players 116. The computing platform 103 may also be used to control mobile digital audio players 115 and fixed digital audio players 116 by changing the current playlist or the currently playing digital audio content, among other things, on the mobile digital audio players 115 or fixed digital audio players 116.

In another embodiment of the invention as illustrated in FIG. 7, the system enables communication between various digital audio players, such as the digital audio players 110-114. This embodiment may be also incorporated with a computing platform 103, for example, acting as a gateway, as discussed above, or alternatively using the computing platform 103 for synchronization among the various digital audio players among the various digital audio players 110-114 or other functions, such as those discussed above.

#### Audio Gateway

FIGS. 4-6 represent exemplary network configuration utilizing different audio gateways for enabling connection of the digital audio players to the Internet or other computer network.

These examples are by no means the only possible configurations that support the invention and do not necessarily cover all aspects of the invention.

**Personal Computer and Digital Audio Players Configuration**

The first exemplary configuration, shown in FIG. 4, uses a personal computer 105 as the audio gateway. The personal computer 105 connects to the Internet or other computer network 102 using a conventional network interface or modem 137. The personal computer 105 is thus able to download digital audio content and other information 101 from the server 100 (FIG. 1) connected to the Internet or other computer network 102. The digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, can then be stored in a persistent storage 133 (FIG. 8), such as a hard drive, on the personal computer 105. The user can also create new playlists using the personal computer 105.

In this embodiment, a wireless access point 106 is used to access the wireless network or proprietary wireless communication platform 104. The wireless network or proprietary wireless communication platform 104 is used by the personal computer 105, acting as the audio gateway, to communicate with mobile digital audio players 115 and fixed digital audio players 116. The personal computer 105, using the wireless network or proprietary wireless communication platform 104, is able to, either automatically or at user request, pass the digital audio content and other information 101, including new playlists, to mobile digital audio players 115 and fixed digital audio players 116. If a fixed digital audio player 116, such as a stand-alone player 112 or a rack player 113 that connects to a stereo 114, happens to be turned off at the time, then the personal computer 105 is able to automatically detect the next time the fixed digital audio player 116 is turned on. When the personal computer 105 detects that a fixed digital audio player 116 has just turned on, then the personal computer 105 can pass the digital audio content and other information 101 to the fixed digital audio player 116 at that time. Mobile digital audio players 115, such as automotive players 110 and handheld players 111, may be out of range of the wireless network or proprietary wireless communication 104 during normal use. When a mobile digital audio player 115 comes into range of the wireless network or proprietary wireless communication 104, the personal computer 105, acting as an audio gateway, can automatically detect the mobile digital audio player 115 and pass the digital audio content and other information 101 at that time.

In addition, the personal computer 105 can, either automatically or upon user request, determine the current playlist and current position within the playlist on a particular mobile digital audio player 115 or fixed digital audio player 116. Then the personal computer 105 can propagate this playlist information to any other mobile digital audio players 115 and fixed digital audio players 116 that are on and in range. This allows a user to move from one mobile digital audio player 115 or fixed digital audio player 116 to another mobile digital audio player 115 or fixed digital audio player 116 and automatically be able to continue the same music and playlist in a seamless manner.

#### **Set-top Box and Digital Audio Players Configuration**

Another exemplary configuration, shown in FIG. 5, uses a set-top box 107 as the audio gateway. The set-top box 107 can connect to the Internet or other computer network 102 either through the same cable or satellite connection that provides the analog or digital audio or video 151 (FIG. 8) that is passed to an audio or video playback device, such as a television set 108, or through an internal or external network interface or modem 137. The set-top box 107 can thus download digital audio content and other information 101 from a server 100 connected to the Internet or other computer network 102. The digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, can then be stored in persistent storage 133, such as a hard drive or flash memory, on the set-top box 107.

In this embodiment, a wireless network interface or proprietary wireless communication interface 141 is used to handle the wireless network or proprietary wireless communication 104. The set-top box 107, acting as the audio gateway, uses the wireless network or proprietary wireless communication 104 to communicate with mobile digital audio players 115 and the fixed digital audio players 116. The set-top box 107, using the wireless network or proprietary wireless communication 104, is able to, either automatically or at user request, pass the digital audio content and other information 101 to mobile digital audio players 115 and fixed digital audio players 116.

If a fixed digital audio player 116, such as a stand-alone player 112 or a rack player 113 that connects to a stereo 114, happens to be turned off at the time, then the set-top box 107 is able to automatically detect the next time the fixed digital audio player 116 is turned on. When the set-top box 107 detects that a fixed digital audio player 116 has just turned on, then the set-

Attorney Docket No. 11748/17

top box 107 can pass the digital audio content and other information 101 to the fixed digital audio player 116 at that time. Mobile digital audio players 115, such as automotive players 110 and handheld players 111, are typically out of range of the wireless network or proprietary wireless communication 104 during normal use.

When a mobile digital audio player 115 comes into range of the wireless network or proprietary wireless communication 104, the set-top box 107, acting as an audio gateway, can automatically detect the mobile digital audio player 115 and pass the digital audio content and other information 101 at that time. In addition, the set-top box 107 can determine, either automatically or upon user request, the current playlist and current position within the playlist on a particular mobile digital audio player 115 or fixed digital audio player 116. Then the set-top box 107 can propagate this playlist information to any other mobile digital audio players 115 and fixed digital audio players 116 that are on and in range. This allows a user to move from one mobile digital audio player 115 or fixed digital audio player 116 to another mobile digital audio player 115 or fixed digital audio player 116 and automatically be able to continue the same music and playlist in a seamless manner.

#### **Stand-alone Gateway and Digital Audio Players Configuration**

Another exemplary configuration, shown in FIG. 6, uses a stand-alone audio gateway 109 as the audio gateway. The stand-alone audio gateway 109 connects to the Internet or other computer network 102 using a network interface or modem 137. The stand-alone audio gateway 109 can download digital audio content and other information 101 from a server 100 connected to the Internet or other computer network 102. The digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, can then be stored in persistent storage 133, such as a hard drive or flash memory, on the stand-alone audio gateway 109. In this embodiment, a wireless network interface or proprietary wireless communication interface 141 (FIG. 8) is used to handle the wireless network or proprietary wireless communication 104. The wireless network or proprietary wireless communication 104 is used by the stand-alone audio gateway 109 to communicate with mobile digital audio players 115 and fixed digital audio players 116. The stand-alone audio gateway 109, using the wireless network or proprietary wireless communication 104, is able to, either automatically or at user request, pass the digital audio content and other information 101 to mobile digital audio players 115 and

Attorney Docket No. 11748/17

fixed digital audio players 116. If a fixed digital audio player 116, such as a stand-alone player 112 or a rack player 113 that connects to a stereo 114, happens to be turned off at the time, then the stand-alone audio gateway 109 is able to automatically detect the next time the fixed digital audio player 116 is turned on. When the stand-alone audio gateway 109 detects that a fixed digital audio player 116 has just turned on, then the stand-alone audio gateway 109 can pass the digital audio content and other information 101 to the fixed digital audio player 116 at that time.

Mobile digital audio players 115, such as vehicle players 110 and portable or handheld players 111, may be out of range of the wireless network or proprietary wireless communication 104 during normal use. When a mobile digital audio player 115 comes into range of the wireless network or proprietary wireless communication 104, the stand-alone audio gateway 109 can automatically detect the mobile digital audio player 115 and pass the digital audio content and other information 101 at that time.

In addition, the stand-alone audio gateway 109 can, either automatically or upon user request, determine the current playlist and current position within the playlist on a particular mobile digital audio player 115 or fixed digital audio player 116. Then the stand-alone audio gateway 109 can propagate this playlist information to any other mobile digital audio players 115 and fixed digital audio players 116 that are on and in range. This allows a user to move from one mobile digital audio player 115 or fixed digital audio player 116 to another mobile digital audio player 115 or fixed digital audio player 116 and automatically be able to continue the same music and playlist in a seamless manner.

#### Local Wireless Network

In another embodiment, shown in FIG. 3, a local wireless network is formed which enables wireless communication between a host, such as a personal computing platform 103 and various digital audio players, such as ~~X~~ mobile digital audio player 115, ~~X~~ digital audio players 116, a stand alone audio gateway 109 and a set top box 107. As shown, various audio gateways are used to establish the network. However, in this embodiment, audio gateways, which contain wireless communication platforms as discussed above, are used primarily for establishing network communication and may or may not be connected to a remote server 100.

Wireless communications between the personal computing platform 103 and mobile digital audio players 115 and fixed digital audio players 116, can be done using industry standard



Attorney Docket No. 11748/17

wireless communications and networking technology, such as Bluetooth, HomeRF, and IEEE 802.11. In addition, with respect to this invention, a proprietary wireless communications technology may also be used for wireless communications. Use of the wireless network or proprietary wireless communication 104 by computing platforms 103, mobile digital audio players 115, and fixed digital audio players 116 may be handled as an internal or external peripheral in the form of a wireless network interface or proprietary wireless communication interface 141. The wireless network or proprietary wireless communication 104 may also require an external wireless access point 106 to handle or facilitate wireless communications and to act as a bridge between the wireless network and wired networking connections, such as may be used by a personal computer 105.

#### Communication Between Digital Audio Players

FIG. 7 illustrates a wireless network configuration which enables communication directly among various digital audio players without a host. The various digital audio players, such as mobile digital audio players 115 and fixed digital audio players 116, use the same wireless network or proprietary wireless communication platform 104 that is used to wirelessly communicate with the computing platform 103, to communicate with each other. The wireless communication between the various digital audio players may be handled by an internal or external wireless communication platform or wireless communication interface 141 (FIG. 8) in each of the disposed digital audio players. In this embodiment, communication between the various digital audio players include directly passing digital audio content and other information 101, including playlists from, for example, one mobile digital audio player 115 or fixed digital audio player 116 to another.

#### Computing and Player Architectures

Figs. 8 and 9 illustrate architectures for the computing platform and stand-alone audio gateway platforms. Figs. 10-15 illustrate the architectures for the various digital audio player platforms. As shown, the architecture of the various platforms is similar. Thus, like reference numbers are used for like components for clarity.

Computing Platform

FIG. 8 illustrates the typical system architecture of a computing platform 103, which can encompass anything from general-purpose devices, such as personal computers 105, to open fixed function devices, such as set-top boxes 107 or stand-alone audio gateways 109, among others. In general, the computing platform 103 has a main processor 130, such as an Intel Pentium III ~~or better~~, for executing various software components. The various software instructions are typically stored in read only memory, or ROM, or flash memory 136, or local storage 132. The local storage 132 can consist of persistent storage 133, such as hard drives or flash memory, or removable storage 134, such as floppy drives, CD-ROM drives, or DVD drives. The software instructions may be executed by the main processor 130 directly from their storage location or loaded into random access memory or RAM 135 to be executed from RAM 135 by the main processor 130. The local storage 132 can also be used to cache digital audio content and other information 101.

The computing platform 103 uses a network interface or modem 137 to access servers 100 on the Internet or other computer network 102, in order to download digital audio content or other information 101. The network interface or modem 137 ~~(JEFF. PROVIDE REFERENCE)~~ may be connected internally or externally to the computing platform 103 using a system bus or peripheral bus 131. The system bus and peripheral buses 131 are provided for connecting internal and external devices to the computing platform 103 in a standard manner. Typical system and peripheral buses 131 include Universal Serial Bus, commonly referred to as USB, IEEE 1394 bus, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI.

The computing platform 103 also supports connection through a user input interface 142 to external or integrated user input devices 153, such as keyboards and mice. In order to provide for output to the user, the computing platform 103 may also contain a display controller 138 ~~(JEFF. PROVIDE REFERENCE)~~, which stores graphical data such as windows, bitmaps and text. The display controller 138 outputs the graphical data in a video output 150 format that is typically displayed to the user on a video monitor, television 108, or LCD panel. In addition to video output 150, the computing platform 103 can provide audio output 152, which is handled by audio playback hardware 140.

for example, on NVIDIA model  
no GeForce 2

for example, a 200M model  
PCI network  
interface  
card,  
Ethernet  
10/100 mb

Attorney Docket No. 11748/17

For a computing platform 103 that is acting as a set-top box 107, the computing platform 103 will likely also contain an analog or digital audio and video decoder 139, for example as disclosed in (PROVIDE REFERENCE), hereby incorporated by reference. The analog or digital audio and video decoder 139 decodes the analog or digital audio or video 151 from sources such as cable or satellite, and passes the audio output 152 and video output 150 to an audio and video playback device, such as a television set 108.

For wireless communication with other computing platforms 103, and various digital audio players, such as mobile digital audio players 115, and fixed digital audio players 116 on a wireless network or proprietary wireless communication 104, the computing platform 103 uses an internal or external wireless network interface or proprietary wireless communication interface 141. ~~(JEFF. PLEASE NOTE WE NEED TO DEFINE PRECISELY ANYTHING THAT IS PROPRIETARY)~~ It should be noted that a computing platform 103 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or features not listed.

### Gateway Platform

FIG. 9 demonstrates some (the unique capabilities of the stand-alone audio gateway 109, though this example is by no means complete or exhaustive in its coverage of the possible options for a stand-alone audio gateway 109. In particular, the stand-alone audio gateway 109 acts as a fixed function device, whose main purpose is to be an audio gateway. The fixed function nature of the stand-alone audio gateway 109 is unlike the personal computer, which exists as a general-purpose computing device. The stand-alone audio gateway 109 is able to connect to the Internet or other computer network 102 using an internal or external network interface or modem 137. The stand-alone audio gateway 109 is able to cache digital audio content and other information 101 downloaded from a server 100 connected to the Internet or other computer network 102 into persistent storage 133, such as a hard drive, on the stand-alone audio gateway 109.

FIG. 9 illustrates a typical system architecture of the stand-alone audio gateway 109. In general, the stand-alone audio gateway 109 has a main processor 130 for executing various software components. The various software components are typically stored in read only memory, or ROM, or flash memory 136, or local storage 132. Local storage 132 can consist of

a C-Cube model no AVIA 6001

Highlight numbers

Attorney Docket No. 11748/17

persistent storage 133, such as hard drives or flash memory, or removable storage 134 such as floppy drives, CD-ROM drives, or DVD drives. The software components are executed by the main processor 130 directly from their storage location or are loaded into random access memory or RAM 135, to be executed from RAM 135 by the main processor 130. Local storage 132 can also be used to cache digital audio content and other information 101. The stand-alone audio gateway 109 uses a network interface or modem 137 to access servers 100 on the Internet or other computer network 102, in order to download digital audio content or other information 101. The network interface or modem 137 is connected internally or externally to the stand-alone audio gateway 109 using a system bus or peripheral bus 131. The system bus and peripheral buses 131 are provided for connecting internal and external devices to the stand-alone audio gateway 109 in a standard manner. Typical system and peripheral buses 131 include Universal Serial Bus, commonly referred to as USB, IEEE 1394, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI. The stand-alone audio gateway 109 also supports connection through a user input interface 142 to external or integrated user input devices 153, such as buttons, keyboards and mice. For output to the user, the stand-alone audio gateway 109 may contain a display controller 138, which stores display data such as windows, bitmaps and text. The display controller 138 outputs the display data in a video output 150 format that is typically displayed to the user on a video monitor, television, or LCD panel. In addition to video output 150, the stand-alone audio gateway 109 can provide audio output 152, which is handled by audio playback hardware 140. For wireless communication with mobile digital audio players 115, and fixed digital audio players 116 on a wireless network or proprietary wireless communication 104, the stand-alone audio gateway 109 uses an internal or external wireless network interface or proprietary wireless communication interface 141. It should be noted that a stand-alone audio gateway 109 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or features not listed.

### Mobile Player

Many different types of mobile digital audio players 115 are suitable for use with the present invention. FIG. 10 demonstrates the general architecture for a mobile digital audio player 115. In general, a mobile digital audio player 115 has a processor 155 that is responsible

## Attorney Docket No. 11748/17

for executing various software and firmware components. The various software and firmware components are typically stored in read only memory, or ROM, or flash memory 158 or in player storage 156, such as a hard drive, flash memory, or removable media. The software and firmware components are executed by the processor 155 directly from their storage location or are loaded into random access memory or RAM 157 to be executed from RAM 157 by the processor 155. Player storage 156 can also be used for storing digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, for later playback and presentation to the user. Typically, the digital audio content 101 is in some encoded format. The audio decoder 162 decodes the digital audio content 101 and passes it to the audio digital to analog converter 163, or DAC. The audio DAC 163 converts the decoded audio to analog and then provides audio output 166 from the mobile digital audio player 115. The audio output 166 of a mobile digital audio player 115 is typically passed to an amplifier or headphones. Communication using a wireless network or proprietary wireless communication 104 by the mobile digital audio player 115 with a ~~stand alone audio gateway 109~~ <sup>computing platform 103,</sup> other mobile digital audio players 115, and fixed digital audio players 116 is done using an internal or external wireless network interface or proprietary wireless communication interface 141. For input from the user, the mobile digital audio player 115 contains user inputs 165, such as buttons or a touch screen. The user input interface 164 handles the actual interface with the user inputs 165, while interpretation of these inputs are typically handled by software and firmware running on the processor 155. For output to the user, the mobile digital audio player 115 may contain a display controller 160, which can provide text and possibly graphical output to the user on an LCD display 161. Tying of the functional components and processor 155 together is typically done using a system bus and peripheral buses 159. Examples of system and peripheral buses 159 include Universal Serial Bus, commonly referred to as USB, IEEE 1394, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI. It should be noted that some of the functional blocks described might encompass multiple physical components. As well, multiple functional blocks may be contained in a single physical component. It should also be noted that a mobile digital audio player 115 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or features not listed.

Attorney Docket No. 11748/17

Fixed Player

There are many different types of fixed digital audio players 116. FIG. 11 demonstrates the general architecture for a fixed digital audio player 116. In general, a fixed digital audio player 116 has a processor 155 that is responsible for executing various software and firmware components. The various software and firmware components are typically stored in read only memory, or ROM, or flash memory 158 or in player storage 156, such as a hard drive, flash memory, or removable media. The software and firmware components are executed by the processor 155 directly from their storage location or are loaded into random access memory or RAM 157 to be executed from RAM 157 by the processor 155. Player storage 156 can also be used for storing digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, for later playback and presentation to the user. Typically, the digital audio content 101 is in some encoded format. The audio decoder 162 decodes the digital audio content 101 and passes it to the audio digital to analog converter 163, or DAC. The audio DAC 163 converts the decoded audio to analog and then provides audio output 167 from the fixed digital audio player 116. The audio output 167 of a fixed digital audio player 116 is typically passed to a stereo, amplifier, speakers or headphones. Communication using a wireless network or proprietary wireless communication 104 by the fixed digital audio player 116 with ~~stand alone audio gateways 109~~ *computing platforms 103*, mobile digital audio players 115, and other fixed digital audio players 116 is done using an internal or external wireless network interface or proprietary wireless communication interface 141. For input from the user, the fixed digital audio player 116 contains user inputs 165, such as buttons or a touch screen. The fixed digital audio player 116 may also receive infrared input 168 from a remote control. The user input interface 164 handles the actual interface with the user inputs 165 and the infrared input 168, while interpretation of these inputs are typically handled by software and firmware running on the processor 155. For output to the user, the fixed digital audio player 116 may contain a display controller 160, which can provide text and possibly graphical output to the user on an LCD display 161. Tying of the functional components and processor 155 together is typically done using a system bus and peripheral buses 159. Examples of system and peripheral buses 159 include Universal Serial Bus, commonly referred to as USB, IEEE 1394, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI. It should be noted that some of the functional blocks described might encompass multiple physical components. As well, multiple

Attorney Docket No. 11748/17

functional blocks may be contained in a single physical component. It should also be noted that a fixed digital audio player 116 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or features not listed.

### Handheld Player

Many different types of mobile digital audio players 115 are suitable for use with the present invention. For example, FIG. 12 illustrates the general architecture for the handheld player 111. In general, the handheld player 111 includes a processor 155 for executing various software and firmware instructions. The various software and firmware instructions may be stored in read only memory, or ROM, or flash memory 158 or in player storage 156, such as a hard drive, flash memory, or removable media. The software and firmware instructions are executed by the processor 155 directly from their storage location or are loaded into random access memory or RAM 157 to be executed from RAM 157 by the processor 155. ~~in P~~ Player storage 156 can also be used for storing digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, for later playback and presentation to the user. Typically, the digital audio content 101 is in some encoded format. The audio decoder 162, ~~(JEFF WE NEED A REFERENCE FOR THIS)~~ decodes the digital audio content 101 and passes it to the audio digital to analog converter 163, or DAC. The audio DAC ~~(JEFF WE NEED A REFERENCE FOR THIS)~~ 163 converts the decoded audio to analog and then provides audio output 166 from the handheld player 111. The audio output 166 of a handheld player 111 may be used to drive headphones.

Communication using a wireless network or proprietary wireless communication 104 by the handheld player 111 with the computing platforms 103, other mobile digital audio players 115, and fixed digital audio players 116 is done using an internal or external wireless network interface or proprietary wireless communication interface 141. For input from the user, the handheld player 111 contains user inputs 165, such as buttons or a touch screen. The user input interface 164 handles the actual interface with the user inputs 165, while interpretation of these inputs are typically handled by software and firmware running on the processor 155. For output to the user, the handheld player 111 may contain a display controller 160 ~~(JEFF WE NEED A REFERENCE FOR THIS)~~, which can provide text and possibly graphical output to the user on

for example, a Texas Instruments  
model no. TLC 320AD77C

for example, a Texas Instruments  
digital signal processor, model  
no. TMS 320VC5416,

an LCD display 161. Tying of the functional components and processor 155 together is typically done using a system bus and peripheral buses 159. Examples of system and peripheral buses 159 include Universal Serial Bus, commonly referred to as USB, IEEE 1394, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI. It should be noted that some of the functional blocks described might encompass multiple physical components. As well, multiple functional blocks may be contained in a single physical component. It should also be noted that a handheld player 111 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or features not listed.

#### Automotive Player

Another type of mobile digital audio player 115 is the automotive player 110, whose general architecture is shown in FIG. 13. In general, the automotive player 110 includes a processor 155 that is responsible for executing various software and firmware instructions. The various software and firmware components are typically stored in read only memory, or ROM, or flash memory 158 or in player storage 156, such as a hard drive, flash memory, or removable media. The software and firmware instructions are executed by the processor 155 directly from their storage location or are loaded into random access memory or RAM 157 to be executed from RAM 157 by the processor 155. ~~the~~ ~~player storage device~~ 156 can also be used for storing digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, for later playback and presentation to the user.

Typically, the digital audio content 101 is in some encoded format. The audio decoder 162 decodes the digital audio content 101 and passes it to the audio digital to analog converter 163 or DAC. The audio DAC 163 converts the decoded audio to analog and then provides audio output 167 from the automotive player 110. The audio output 167 of an automotive player 110 typically feeds a conventional audio amplifier, which then drives the car speakers. Communication using a wireless network or proprietary wireless communication 104 by the automotive player 110 with computing platforms 103, other mobile digital audio players 115, and fixed digital audio players 116 is done using an internal or external wireless network interface or proprietary wireless communication interface 141.



Attorney Docket No. 11748/17

For input from the user, the automotive player 110 contains user inputs 165, such as buttons or a touch screen. The user input interface 164 handles the actual interface with the user inputs 165, while interpretation of these inputs are typically handled by software and firmware running on the processor 155. In addition, an automotive player 110 may support voice commands for user input. If voice commands are supported, a microphone 174 is used to feed analog audio to the audio analog to digital converter 173, which converts the analog audio to digital. Then, the audio capture hardware 172 and the processor <sup>155</sup> will interpret the voice commands from the user. For output to the user, the automotive player 110 may contain a display controller 160, which can provide text and possibly graphical output to the user on an LCD display 161. Tying of the functional components and processor 155 together may be accomplished by way of a system bus and peripheral buses 159. Examples of suitable system and peripheral buses 159 include Universal Serial Bus, commonly referred to as USB, IEEE 1394, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI.

It should be noted that some of the functional blocks described might encompass multiple physical components. As well, multiple functional blocks may be contained in a single physical component. It should also be noted that an automotive player 110 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or features not listed.

### Rack Player

There are many different types of fixed digital audio players 116. FIG 14 demonstrates the general architecture for a rack player 113. In general, a rack player 113 includes a processor 155 that is responsible for executing various software and firmware instructions. The various software and firmware instructions may be stored in read only memory, or ROM, or flash memory 158 or in player storage 156, such as a hard drive, flash memory, or removable media. The software and firmware instructions may be executed by the processor 155 directly from their storage location or loaded into random access memory or RAM 157 to be executed from RAM 157 by the processor 155. ~~The~~ <sup>the</sup> Player storage device 156 can also be used for storing digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, for later playback and presentation to the user. Typically, the digital audio content 101

is in some encoded format. The audio decoder 162 decodes the digital audio content 101 and passes it to the audio digital to analog converter 163, or DAC. The audio DAC 163 converts the decoded audio to analog and then provides audio output 167 from the rack player 113. The audio output 167 of a rack player 113 typically is passed to a stereo system 114. Communication using a wireless network or proprietary wireless communication 104 by the rack player 113 with computing platforms 103, mobile digital audio players 115, and other fixed digital audio players 116 is done using an internal or external wireless network interface or proprietary wireless communication interface 141. For input from the user, the rack player 113 contains user inputs 165, such as buttons or a touch screen. The rack player 113 may also receive infrared input 168 from a remote control. The user input interface 164 handles the actual interface with the user inputs 165 and the infrared input 168, while interpretation of these inputs are typically handled by software and firmware running on the processor 155. For output to the user, the rack player 113 may contain a display controller 160, which can provide text and possibly graphical output to the user on an LCD display 161. Tying connection of the functional components and processor 155 together may be accomplished by way of a system bus and peripheral buses 159. Examples of suitable system and peripheral buses 159 include Universal Serial Bus, commonly referred to as USB, IEEE 1394, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI.

It should be noted that some of the functional blocks described might encompass multiple physical components. As well, multiple functional blocks may be contained in a single physical component. It should also be noted that a rack player 113 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or features not listed.

#### **Stand-alone Player**

Another type of fixed digital audio players 116 is the stand-alone player 112, whose general architecture is shown in FIG. 15. In general, a stand-alone player 112 includes a processor 155 that is responsible for executing various software and firmware instructions. The various software and firmware components are typically stored in read only memory, or ROM, or flash memory 158 or in player storage 156, such as a hard drive, flash memory, or removable media. The software and firmware components are executed by the processor 155 directly from

✓ their storage location or are loaded into random access memory or RAM 157 to be executed from RAM 157 by the processor 155. ~~the~~ ~~player~~ storage device 156 can also be used for storing digital audio content and other information 101, such as artists, track names, album names, lyrics, and playlists, for later playback and presentation to the user. Typically, the digital audio content 101 is in some encoded format. The audio decoder 162 decodes the digital audio content 101 and passes it to the audio digital to analog converter 163, or DAC. The audio DAC 163 converts the decoded audio to analog. The analog audio from a stand-alone player 112 typically directly drives speakers 170 attached to the stand-alone player 112. Communication using a wireless network or proprietary wireless communication 104 by the stand-alone player 112 with computing platforms 103, mobile digital audio players 115, and other fixed digital audio players 116 is done using an internal or external wireless network interface or proprietary wireless communication interface 141. For input from the user, the stand-alone player 112 contains user inputs 165, such as buttons or a touch screen. The stand-alone player 112 may also receive infrared input 168 from a remote control. The user input interface 164 handles the actual interface with the user inputs 165 and the infrared input 168, while interpretation of these inputs are typically handled by software and firmware running on the processor 155. For output to the user, the stand-alone player 112 may contain a display controller 160, which can provide text and possibly graphical output to the user on an LCD display 161. Connection of the functional components and processor 155 together is typically done using a system bus and peripheral buses 159. Examples of suitable system and peripheral buses 159 include Universal Serial Bus, commonly referred to as USB, IEEE 1394, commonly referred to as FireWire, and Peripheral Connect Interface, commonly referred to as PCI.

It should be noted that some of the functional blocks described might encompass multiple physical components. As well, multiple functional blocks may be contained in a single physical component. It should also be noted that a stand-alone player 112 is not limited to the capabilities and features listed in this description, but may contain a subset of the described features or may contain additional capabilities or features not listed.

#### Audio Gateway Software

FIGS. 16 to 20 provide flow diagrams for the audio gateway embodiment of this invention. In these flow diagrams, the software is assumed to be running in a multitasking

## Attorney Docket No. 11748/17

environment, with each of the flow diagrams representing a particular independently running task or process. However, it should be noted that these flow diagrams represent only one of many different ways to implement the key software functionality for the audio gateway and that many other implementations are possible, including those which do not require a multitasking environment.

**Audio Gateway Message Handling Flow**

FIG. 16 provides the flow diagram of the message handler for the audio gateway. In general, the message handler takes the messages received from other computing platforms 103, mobile digital audio players 115, and fixed digital audio players 116 on the wireless network or proprietary wireless communication 104 and queues these messages for use by other processes or handles them itself, depending on the message type. In this embodiment, the message handler is a continuously running process. The block "Start 200" represents the beginning of the message handling process. The message handler checks if there is a message received step 201.

*step* If a message has been received, the message handler then checks to see what type of message it is, among many possible types, as indicated in steps 202-212. After the message handler determines the type of message, an appropriate response is queued and the system returns to step 201 and checks for additional messages. If the message is a broadcast response message from a player 202, then the message handler queues the broadcast response message 203. *X* If the message is a query response message from a player 204, then the message handler queues the query response message in step 205. If the message is a poll response message from a player 206, then the message handler queues the poll response message in step 207. If the message is a playlist response message from a player 208, then the message handler queues the playlist response message in step 209. If the message is a content response message from a player 210, then the message handler queues the content response message in step 211. If the message is a content acknowledge message from a player 212, then the message handler queues the content acknowledge message in step 213. ~~After messages are handled or~~ If the message was none of those previously checked for, the message handler handles or queues any other messages as necessary 214.

Audio Gateway Discovery Flow

Discovery of mobile digital audio players 115 and fixed digital audio players 116 within range of the audio gateway, on the wireless network or proprietary wireless communication 104, is an important capability with respect to this invention. FIG. 17 provides the flow diagram for discovery by the audio gateway of mobile digital audio players 115 and fixed digital audio players 116. In this example, the audio gateway discovery handler is a continuously running process. The step start 220 represents the beginning of the discovery handling process. In order to get a message response from the mobile digital audio players 115 and fixed digital audio players 116, the discovery handler sends a broadcast for players message in step 221. The discovery handler then waits, with a timeout, for example, <sup>5 seconds</sup> (~~JEFF FULL IN~~) for a broadcast response message from any players in step 222. The discovery handler then checks if there is a player broadcast response message in the queue in step 223. If there is no response, then the discovery handler broadcasts again for players. If there is a response, then the discovery handler sends a query player message to a responding player in step 224 to get information about the type of player that has responded. The discovery handler then waits, with some timeout, for a player query response message in step 225 from the player that previously responded to the broadcast. The discovery handler then checks if there is a query response message in the queue in step 226. If there is no response, then the discovery handler broadcasts again for players. If there is a response, then the discovery handler checks the information returned in the query response message to see if the player is already known in step 227. If the player is already known, then the discovery handler broadcasts again for players. However, a player is unlikely to respond to a broadcast from an audio gateway when the player and audio gateway already know about each other. If the player is not already known, then the discovery handler adds the player to the list of players in proximity in step 228 of the audio gateway. Finally, the discovery handler flags the new player in proximity for playlist continuation in step 229 and for content synchronization in step 230. This allows the playlist continuation handler in the audio gateway to capture the current playlist and current selection from this new player for possible broadcast to other players. Also, this allows the content synchronization handler in the audio gateway to automatically download digital audio content and other information 101 cached on the audio gateway to the new player.

Audio Gateway Dropout Detection Flow

The flow diagram for audio gateway detection of dropout of players is shown in FIG. 18. The dropout detection handler in the audio gateway polls players that are known to be in proximity in order to see if any of the players has possibly gone out of range of the wireless network or proprietary wireless communication 104 or has been turned off. In this example, the dropout detection handler is a continuously running process. Start 240 represents the beginning of the dropout detection handling process. The dropout detection handler checks the list of players in proximity 241 maintained by the audio gateway. If there are players in proximity as determined in step 242, then the dropout detection handler sends a poll message to the next player in proximity in the proximity list in step 243. This allows all the players in the list of players in proximity to be checked in a sequential manner. Then the dropout detection handler waits, with some timeout, for a poll response message from the player in step 244 that was sent the poll message in step 243. If there is no poll response message from the player in the queue in step 245 then the dropout detection handler checks if the player is already flagged as possibly being out of range in step 246 of the wireless network or proprietary wireless communication 104. If the player is not already flagged as possibly out of range 246, then the dropout detection handler flags that the player is possibly out of range in step 247 and checks the list of players in proximity again. If the player is already flagged as possibly out of range in step 246, then the dropout detection handler removes the player from the list of players in proximity in step 248 and checks the list of players in proximity in step 241 again. If the player poll response message is in the queue in step 245, then the dropout detection handler clears the possibly out of range flag in step 249 for the player in the list of players in proximity. Next, the dropout detection handler checks if the player is requesting content synchronization in step 250, based on information passed in the poll response message from the player. If the player is requesting content synchronization, then the dropout detection handler flags the player for content synchronization in step 251 in the list of players in proximity. The content synchronization handler uses this information when deciding which players to update for digital audio content and other information 101. Once the player is flagged for content synchronization or the player is not requesting content synchronization, then the dropout detection handler checks if the player is requesting playlist continuation in step 252, based on information passed in the poll response message from the player. If the player is requesting playlist continuation, then the dropout

The  
Step

Attorney Docket No. 11748/17

detection handler flags the player for playlist continuation <sup>in step</sup> 253 in the list of players in proximity. The playlist continuation handler uses this information when deciding which players to update the playlist and current selection for. Once the player is flagged for playlist continuation in step 253 or the player is not requesting playlist continuation in step 252, then the dropout detection handler checks the list of players in proximity in step 241 again.

### Audio Gateway Content Synchronization Flow

The flow diagram for audio gateway content synchronization is shown in FIG. 19, with content synchronization being a key capability of the invention. The content synchronization handler in the audio gateway checks for players that need content synchronization. Content synchronization involves updating or adding digital audio content and other information 101 to a player when the audio gateway has digital audio content and other information 101 that is not contained on the player. This may be handled automatically when the player has recently been discovered as being in proximity by the gateway discovery handler or the player directly requests content synchronization through poll response messages to the gateway. In this example, the content synchronization handler is a continuously running process. The step "Start 260" represents the beginning of the content synchronization handling process. The content synchronization handler checks the list of players in proximity in step 261 maintained by the gateway. If there are players in proximity flagged for content synchronization in step 262, then the content synchronization handler sends a query player for content message to the player in step 263 that is flagged for content synchronization. Next, the content synchronization handler waits, with some timeout, for a player content response message in step 264. If there is no content response message in the queue in step 265 from the player that was sent the query player for content message in step 263, then the content synchronization handler clears the content synchronization flag for the player in the proximity list in step 266 and checks the list of players in proximity again. If there is a content response message in the queue <sup>in step 265</sup> from the player that was sent the query player for content message, then the gateway compares the digital audio content in the player with the digital audio content in the gateway in step 267. The player's digital audio content information is contained in the content response message sent to the gateway by the player. Next, the content synchronization handler checks if there is any content in the gateway that is not on the player in step 268. If the player content is properly synchronized with the

## Attorney Docket No. 11748/17

gateway, then the content synchronization handler clears the content synchronization flag for the player in the proximity list and checks the list of players in proximity in step 261 again. If there is content on the gateway that is not on the player in step 268, then the content synchronization handler checks if there is storage on the player for the new content in step 269. The available storage on the player is provided in the content response message that the player sent to the gateway. If there is not sufficient storage on the player for the new content in step 269, then the content synchronization handler clears the content synchronization flag for the player in the proximity list in step 266 and checks the list of players in proximity in step 261 again. If there is storage on the player for the new content as determined in step 269, then the content synchronization handler sends the content data to the player in step 270. Next, the content synchronization handler waits, with some timeout, for the content acknowledge message from the player in step 271. If there is no content acknowledge message in the queue in step 272, then the content synchronization handler clears the content synchronization flag for the player in the proximity list in step 266 and checks the list of players in proximity again. <sup>in step 261</sup> If there is a content acknowledge message in the queue from the player, then the content synchronization handler checks to see, from the compare of content in the player with content in the gateway, if there is more content to send to the player in step 273. If there is more content to send to the player then the content synchronization handler checks again if there is storage on the player for the new content in step 269, and so on until there is no more content to pass from the gateway to the player. If there is no more content to send to the player, then the content synchronization handler clears the content synchronization flag for the player in the proximity list in step 266 and checks the list of players in proximity in step 261 again.

**Audio Gateway Playlist Continuation Flow**

The flow diagram for audio gateway playlist continuation is shown in FIG. 20, with playlist continuation being a key capability of the invention. The playlist continuation handler in the audio gateway checks for propagation of the playlist and current playlist selection from one mobile digital audio player 115 or fixed digital audio player 116 to all other mobile digital audio players 115 and fixed digital audio players 116 in proximity. Playlist continuation involves seamless continuation of playback of digital audio content 101 from a particular playlist as a user moves from one mobile digital audio player 115 or fixed digital audio player 116 to another.



## Attorney Docket No. 11748/17

This may be handled automatically when the gateway discovery handler discovers a player as being in proximity, where the player is currently playing digital audio content 101. The player itself may also directly request playlist continuation through poll response messages to the gateway.

✓ In this example, the playlist continuation handler is a continuously running process. The step "Start" ~~280~~ represents the beginning of the playlist continuation handling process. The playlist continuation handler checks the list of players in proximity in step 281 maintained by the gateway. If there are players in proximity flagged for playlist continuation in step 282, then the playlist continuation handler sends a query player for playlist message to the player in step 283 that is flagged for playlist continuation. Next, the playlist continuation handler waits, with some timeout, for a player playlist response message in step 284. If there is no playlist response message in the queue ~~285~~ from the player that was sent the query player for playlist message *in step* 283, then the playlist continuation handler clears the playlist continuation flag for the player in the proximity list in step 286 and checks the list of players in proximity again. If there is a playlist response message in the queue as determined in step 285 from the player that was sent the query player for playlist message in step 283, then the gateway checks the playlist response message to see if the playlist and current position within the playlist, both of which are contained in the playlist response message, are valid in step 287. If the playlist and current position are not valid, then the playlist continuation handler clears the playlist continuation flag for the player in the proximity list and checks the list of players in proximity again. If the playlist and current position in the playlist are valid as determined in step 287, then the playlist continuation handler checks the list of players in proximity in step 288. If there are any other players in proximity as determined in step 289, then the playlist continuation handler sends a broadcast playlist and current position message to all other players in proximity in step 290. After the playlist continuation handler sends a broadcast playlist and current position message to all other players in proximity or if there are no other players in proximity, then the playlist continuation handler clears the playlist continuation flag for the player in the proximity list in step 286 and checks the list of players in proximity in step 281 again.

*in step*

*in step 286*

*in step 286*

Player Software

FIGS. 21-27 provide flow diagrams for the various digital audio players. In these flow diagrams, the software is assumed to be running in a multitasking environment, with each of the flow diagrams representing a particular independently running task or process. However, it should be noted that these flow diagrams represent only one of many different ways to implement the key software functionality for the player and that many other implementations are possible, including those which do not require a multitasking environment.

Player Message Handling Flow

FIG. 21 is a flow diagram of the message handler for a player. In general, the message handler takes the messages received from computing platforms 103 acting as audio gateways and from other mobile digital audio players 115 and fixed digital audio players 116, on a wireless network or proprietary wireless communication 104, and queues these messages for use by other processes or handles them itself, depending on the message type. In this example, the message handler is a continuously running process. The step "Start 300" represents the beginning of the message handling process. The message handler checks if there is a message received in step 301. If there is a message received, the message handler then checks to see what type of message it is, among many possible types.

After the message handler determines the type of message an appropriate response is queued and the system returns to step 301 and checks for additional messages. If the message is a broadcast for players message from a gateway as determined in step 302, then the message handler queues the broadcast for players message in step 303. After the message handler queues the broadcast for players message, the message handler checks for more messages. If the message is a query player message from a gateway as determined in step 304, then the message handler queues the query player message in step 305. After the message handler queues the query player message, the message handler checks for more messages. If the message is a poll message from a gateway as determined in 306, then the message handler queues the poll message in step 307. After the message handler queues the poll message, the message handler checks for more messages. If the message is a query player for content message from a gateway as determined in 308, then the message handler queues the query player for content message in step 309. After the message handler queues the query player for content message, the message

*Handwritten notes:*  
 "in step 305" (under "the message handler queues the query player message")  
 "in step 307" (under "the message handler queues the poll message")  
 "in step 309" (under "the message handler queues the query player for content message")  
 "in step 309" (at the end of the paragraph)

## Attorney Docket No. 11748/17

handler checks for more messages. If the message is content data from a gateway, then the message handler stores the content in local player storage in step 311. The message handler also sends a content acknowledge message to the gateway in step 312. After the message handler sends a content acknowledge message to the gateway, <sup>in step 312</sup> the message handler checks for more messages. If the message is a query player for playlist message from a gateway as determined in <sup>✓</sup> ~~step~~ 313, then the message handler queues the query player for playlist message in step 314. After the message handler queues the query player for playlist message in step 314, the message handler checks for more messages. If the message is a broadcast playlist message from a gateway as determined in ~~315~~, then the message handler queues the broadcast playlist message in step 316. After the message handler queues the broadcast playlist message, <sup>in step 316</sup> the message handler checks for more messages. Finally, if the message was none of those previously checked for, the message handler handles or queues any other messages as necessary in step 317 and then the message handler checks for more messages.

Player Discovery Flow

Discovery by the audio gateway of mobile digital audio players 115 and fixed digital audio players 116 is an important capability with respect to this invention. FIGS. 22 and 23 provide the flow diagrams for discovery responses by the player when the player detects discovery attempts by an audio gateway. In this example, the player discovery broadcast response handler and the player discovery query response handler are continuously running processes. The step "Start" ~~320~~ <sup>✓</sup> represents the beginning of the discovery broadcast response handling process. The discovery broadcast response handler first checks for a broadcast for players message in the queue in step 321 from a gateway. If there is a broadcast for players message in the queue as determined in 322, then the discovery broadcast response handler checks if the gateway is already in proximity of the player in step 323. The discovery broadcast response handler is able to get information about the gateway from the broadcast for players message received from the gateway and can compare that information with information saved by the discovery query response handler for any gateway in proximity. If the gateway is not already in proximity as determined in step 323, then the discovery broadcast response handler sends a broadcast acknowledge message to the gateway in step 324. After the discovery broadcast response handler sends the broadcast acknowledge message to the gateway in step 324, or if the

## Attorney Docket No. 11748/17

gateway is already in proximity as determined in step 323, or if there is no broadcast for players message in the queue as determined in step 322, then the discovery broadcast response handler checks for a broadcast for players message in the queue ~~X~~ again. ✓

✓ The step "Start" ~~330~~ (FIG. 23) represents the beginning of the discovery query response handling process. The discovery query response handler first checks for query player messages from a gateway in the queue in step 331. If there is a query player message in the queue as determined in ~~332~~, then the discovery query response handler sends a query response message to the gateway in step 333 that sent the query player message. Then the discovery query response handler saves that the gateway is in proximity in step 334 from information obtained from the query player message from the gateway. After the discovery query response handler saves that the gateway is in proximity as determined in step 334 or if there is no query player message in the queue as determined ~~332~~, then the discovery query response handler checks for a query player message from a gateway in the queue again.

step 334

step 332

Player Dropout Detection Flow

The flow diagram for player dropout detection of an audio gateway is shown in FIG. 24. The dropout detection handler in the player watches for poll messages from an audio gateway in order to see if the player has gone out of range of the gateway. In this example, the player dropout detection handler is a continuously running process. Step "Start" ~~340~~ represents the beginning of the player dropout detection handling process. The player dropout detection handler checks if the player is in proximity of a gateway in step 341. The player discovery query response handler, shown in FIG. ~~24~~, saves information about a gateway that is in proximity. If the player is not in proximity of a gateway as determined in step 341, then the player discovery query response handler just continues to check if the player is in proximity of a gateway. If the player is in proximity of a gateway as determined in step 341, then the player discovery query response handler waits, with some timeout, for a poll response message from the gateway in step 342 that is in proximity. The timeout period is significantly more than the polling period used by the gateway. If there is not a poll message in the queue as determined in step 343 from the gateway that is in proximity, then the player discovery query response handler checks if the gateway is already flagged as possibly out of range in step 344. If the gateway is not already flagged as possibly out of range as determined in step 344, then the player discovery query

step 341

23

Attorney Docket No. 11748/17

*in* response handler flags that the gateway is possibly out of range in step 345 and <sup>*then*</sup> continues to check if the player is in proximity of a gateway and returns to step 341. If the gateway is already flagged as possibly out of range as determined in step 344, then the player discovery query response handler removes the gateway as being in proximity in step 346 and then continues to check if the player is in proximity of a gateway in step 341. If there is a poll message in the queue as determined in 343 from the gateway that is in proximity, then the player discovery query response handler checks if the user requested content synchronization of the player in step 347. If the user did request content synchronization of the player as determined in step 347, then the player discovery query response handler flags a content synchronization request in the poll response message in step 348 to the gateway in proximity. If the user did not request content synchronization of the player, then the player discovery query response handler skips flagging of content synchronization in the poll response message in step 348. Next, the player discovery query response handler checks if the user requested playlist continuation for the player in step 349. If the user did request playlist continuation for the player as determined in step 349, then the player discovery query response handler flags a playlist continuation request in the poll response message in step 350 to the gateway in proximity. If the user did not request playlist continuation for the player as determined in step 349, then the player discovery query response handler skips flagging of playlist continuation in the poll response message in step 350. Next, the player discovery query response handler sends the poll response message to the gateway in step 351 that is in proximity and sent the poll message. Next, the player discovery query response handler clears the gateway possibly out of range flag in step 352 if it was set for the gateway in proximity. Then the player discovery query response handler continues to check if the player is in proximity of a gateway as determined in step 341. ✓

#### Player Content Synchronization Flow

✓ The flow diagram for player content synchronization response is shown in FIG. 25, with content synchronization being a key capability of the invention. The content synchronization response handler in the player responds to content queries from a gateway. In this example, the content synchronization response handler is a continuously running process. The step "Start 360" represents the beginning of the content synchronization response handling process. The content synchronization response handler checks for a query player content message in the queue in step

Attorney Docket No. 11748/17

361 from a gateway. If there is a query player content message in the queue as determined in *step* ✓ 362, then the content synchronization response handler builds a content response message by first getting a list of all the digital audio content on the player in 363. Next, the content synchronization response handler determines the amount of available storage space on the player in step 364 for additional digital audio content. Finally, the content synchronization response handler sends a player content response message in step 365 to the gateway that sent the query player content message. The player content response message contains the list of all the digital audio content on the player as well as the amount of available space on the player. Once the content synchronization response handler sends a player content response message *as determined* to the gateway that sent the query player content message or there is no query player content message in the queue in step 362, then the content synchronization response handler checks for a query player content message in the queue again. *for step 365*

#### Player Playlist Continuation Flow

FIGS. 26 and 27 represent flow diagrams for playlist continuation response and playlist continuation updating by the player when the player detects playlist continuation query and updating attempts by an audio gateway. Playlist continuation is a key capability of the invention.

✓ In this example, the player playlist response handler and the player playlist update handler are continuously running processes. The step "Start" ~~380~~ represents the beginning of the playlist response handling process. First, the playlist response handler checks for a query player for playlist message in the queue in step 381 from a gateway in proximity. If there is a query player for playlist message in the queue as determined in step 382, then the playlist response handler gets the current playlist and current position within the playlist in step 383 and puts this information in a playlist response message. Next, the playlist response handler sends the playlist response message to the gateway in step 384 that sent the query player for playlist message. After the playlist response handler sends the playlist response message to the gateway as determined 384 or there is not a query player for playlist message in the queue as determined in step 382, then the playlist response handler checks for a query player for playlists message 381 in the queue again.

✓ The "Start" ~~390~~ (FIG. 27) represents the beginning of the playlist update handling process. First the playlist update handler checks for a broadcast playlist message in step 391 in

## Attorney Docket No. 11748/17

the queue from a gateway in proximity. If there is not a broadcast playlist message in the queue as determined in step 392, then the playlist update handler just checks for a broadcast playlist message in the queue again. If there is a broadcast playlist message in the queue, then the playlist update handler checks if the playlist already exists on the player in step 393. The playlist information is found in the broadcast playlist message. If the playlist already exists on the player, then the playlist update handler activates the playlist and sets the current position within the playlist in step 394 on the player. The current position within the playlist is found in the broadcast playlist message. Then the playlist update handler checks for a broadcast playlist message in step 391 in the queue again. If the playlist does not already exist on the player as determined in step 393, then the playlist update handler saves the new playlist on the player in step 395. Next, the playlist update handler checks if the player is currently playing in step 396. If the player is not currently playing, then the playlist update handler sets the new playlist as the current playlist in step 397 and sets the current position within the playlist in step 394. If the player is currently playing, then the playlist update handler notifies the user that a new playlist is available in step 398. This allows the user to decide to play the new playlist or continue with a current playlist. Next, the playlist update handler checks for a broadcast playlist message 391 in the queue again.

as  
determined  
in  
step  
392  
as  
determined  
in  
step  
393  
in  
step

Obviously, many modifications and variations of the present invention are possible in light of the above teachings. Thus, it is to be understood that, within the scope of the appended claims, the invention may be practiced otherwise than as specifically described above.

What is claimed and desired to be covered by a Letters Patent is as follows:

Attorney Docket No. 11748/17

ABSTRACT OF THE INVENTION

A wireless communication system and in particular to a wireless communication system for digital audio players that provides for increased functionality, such as communication, interaction and synchronization between a computing platform and various mobile, portable or fixed audio digital players as well as providing a communication link between the various digital audio players themselves. The computing platform may act, for example, through a wireless communication platform, to control the digital audio players; to act as a cache of digital audio data for the digital audio players; as well as provide a gateway to the Internet to enable the digital audio players to access additional digital audio content and other information. The computing platform may also be used to automatically update digital audio content on the digital audio players; synchronize digital audio content and playlist between digital audio players; and automatically continue a particular playlist as the user moves <sup>from</sup> one audio digital player to another.



Attorney Docket No. 11748/17

IN THE CLAIMSWe Claim:

1. A local wireless communication network for digital audio players, the network comprising:

a host;

a first wireless communication platform coupled to said host; and

one or more digital audio players having a second wireless communication platform coupled to each of said one or more digital audio players for establishing a communication link between said host and said one or more digital audio players.

2. The network as recited in claim 1, wherein said host and said first wireless communication platform are configured as a gateway.

3. The network as recited in claim 2, wherein said host is a personal computing platform.

4. The network as recited in claim 2, wherein said host is a set top box.

5. The network as recited in claim 2, wherein said host is a stand alone audio gateway.

6. A local wireless communication network for digital audio players, the network comprising:

one or more digital audio players; and

one or more wireless communication platforms, said one or more wireless communication platforms coupled to said one or more digital audio players for establishing a communication link between at least two of said one or more digital audio players.

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**